Fast Optimization with Zeroth-Order Feedback in Distributed, Multi-User MIMO Systems

Olivier Bilenne, Panayotis Mertikopoulos, Elena Veronica Belmega

Fast Optimization with Zeroth-Order Feedback in Distributed, Multi-User MIMO Systems

Olivier Bilenne, Panayotis Mertikopoulos, *Member, IEEE* and E. Veronica Belmega, *Senior Member, IEEE*

***Abstract*—In this paper, we develop a gradient-free optimization methodology for efficient resource allocation in Gaussian MIMO multiple access channels. Our approach combines two main ingredients: (*i*) an entropic semidefinite optimization based on matrix exponential learning (MXL); and (*ii*) a one-shot gradient estimator which achieves low variance through the reuse of past information. This novel algorithm, which we call *gradient-free MXL with callbacks* (MXL0$^+$), retains the convergence speed of gradient-based methods while requiring minimal feedback per iteration—a *single* scalar. In more detail, in a MIMO multiple access channel with K users and M transmit antennas per user, the MXL0$^+$ algorithm achieves ε-optimality within $\mathrm{poly}(K,M)/\varepsilon^2$ iterations (on average and with high probability), even when implemented in a fully distributed, asynchronous manner. For cross-validation, we also perform a series of numerical experiments in medium- to large-scale MIMO networks under realistic channel conditions. Throughout our experiments, the performance of MXL0$^+$ matches—and sometimes exceeds—that of gradient-based MXL methods, all the while operating with a vastly reduced communication overhead. In view of these findings, the MXL0$^+$ algorithm appears to be uniquely suited for distributed massive MIMO systems where gradient calculations can become prohibitively expensive.**

***Index Terms*—Gradient-free optimization; matrix exponential learning; multi-user MIMO networks; throughput maximization.**

I. Introduction

THE deployment of multiple-input and multiple-output (MIMO) terminals at a massive scale has been identified as one of the key enabling technologies for fifth generation (5G) wireless networks, and for good reason: massive-MIMO arrays can increase throughput by a factor of 10× to 100× (or more), they improve the system's robustness to ambient noise and channel fluctuations, and they bring about significant latency reductions over the air interface [1, 2]. Moreover, ongoing discussions for the evolution of 5G envision the deployment of advanced MIMO technologies at an even larger scale in order to reach the throughput and spectral efficiency required for "speed of thought" connectivity [3, 4].

In view of this, there have been intense efforts to meet the complex technological requirements that the massive-MIMO paradigm entails. At the hardware level, this requires scaling up existing multiple-antenna transceivers through the use of inexpensive service antennas and/or time-division duplexing (TDD) [1, 5, 6]. At the same time however, given the vast amount of resources involved in upgrading an ageing infrastructure, a brute-force approach based solely on the evolution of wireless hardware technology cannot suffice. Instead, unleashing the full potential of massive-MIMO arrays requires a principled approach with the aim of minimizing computational overhead and related expenditures as the network scales up to accommodate more and more users.

In this general multi-user MIMO context, it is crucial to optimize the input signal covariance matrix of each user, especially in the moderate (or low) signal to interference-plus-noise ratio (SINR) regime [7–11]. The conventional approach to this problem involves the use of water-filling (WF) solution methods, either *iterative* (IWF) [8, 12] or *simultaneous* (SWF) [13]. In the IWF algorithm only one transmitter updates its input covariance matrix per iteration (selected in a round-robin fashion); instead, in SWF all transmitters update their transmission characteristics simultaneously. Owing to this "parallelizability", SWF can be deployed in a distributed and decentralized fashion; on the other hand, because of potential clashes in the users' concurrent updates, the SWF algorithm may fail to converge [13]. By comparison, IWF *always* converges to an optimal state [8], but this comes at the cost of centralization (to orchestrate the updating transmitters at each iteration) and a greatly reduced convergence speed (which is inversely proportional to the number of users in the system).[1]

In addition to the above, the authors of [15] proposed the so-called *iterative weighted MMSE* (IW-MMSE) algorithm to solve the (non-convex) throughput maximization problem in the broadcast channel (downlink). This work was subsequently extended in [16] to broadcasting in multi-cell interference channels. This formulation includes as a special case the uplink multiple access channel (MAC) under the assumption that (*a*) all receivers are co-located and act as a single entity; and (*b*) this amalgamated entity employs successive interference cancellation (SIC) to decode incoming messages. In this context, IW-MMSE was shown to converge to an optimal solution in a distributed fashion, without suffering the convergence/distributedness trade-off of water-filling methods.

Importantly, the above schemes rely on each user having perfect knowledge of (*a*) their effective channel matrix (which typically changes from one transmission frame to another); and/or (*b*) the global, system-wide signal-plus-noise covariance

O. Bilenne and P. Mertikopoulos are with Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, 38000 Grenoble, France; P. Mertikopoulos is also with Criteo AI Lab, Grenoble, France. E. V. Belmega is with ETIS, CY Cergy Paris University, ENSEA, CNRS, UMR 8051, F-95000, Cergy, France.

The authors are grateful for financial support from the French National Research Agency (ANR) projects ORACLESS (ANR–16–CE33–0004–01) and ELIOT (ANR–18–CE40–0030 and FAPESP 2018/12579–7). This research has also received financial support from the COST Action CA 16228 'European Network for Game Theory' (GAMENET).

[1] As suggested by one of the referees, it is worth pointing out here that water-filling has also been applied to a broad range of distributed network paradigms; see e.g., [14] for an application to cognitive radio OFDM networks.

Algorithm	[source]	Feedback	Convergence	Conv. Speed	Distributed	Overhead
IWF	[8]	full matrix	✓	$\mathcal{O}(K\log(1/\varepsilon))$	no	$\mathcal{O}(\min\{M^2, N^2\})$
SWF	[13]	full matrix	no	—	✓	$\mathcal{O}(\min\{KM^2, N^2\})$
IW-MMSE	[15, 16]	full matrix	✓	—	✓	$\mathcal{O}(\min\{KM^2, N^2\})$
MXL	[17]	full matrix (imp.)	✓	$\mathcal{O}(1/\varepsilon^2)$	✓	$\mathcal{O}(\min\{KM^2, N^2\})$
MXL0	[this paper]	scalar	✓	$\mathcal{O}(1/\varepsilon^4)$	✓	$\mathcal{O}(1)$
MXL0$^+$	[this paper]	scalar	✓	$\mathcal{O}(1/\varepsilon^2)$	✓	$\mathcal{O}(1)$

TABLE I: Overview of related work. For the purposes of this table "full matrix feedback" refers to the case where the network's users have perfect knowledge of *a*) their effective channel matrices; and/or *b*) the aggregate signal-plus-noise covariance matrix at the receiver at each transmission frame. The characterization "imp." (for "imperfect") signifies that noisy measurements suffice; on the contrary, "scalar" means that users only observe their *realized* utility (in our case, their achieved throughput). The "convergence" and "conv. speed" columns indicate the best theoretical guarantees for each algorithm: $f(\varepsilon)$ denotes the maximum number of iterations required to reach an ε-optimal state while "—" means that no guarantees are known. Finally, the "overhead" column indicates the computation/communication overhead of each iteration; here and throughout, K is the number of users, M is the maximum number of transmit antennas per user, and N is the number of antennas at the receiver.

matrix at the receiver. These elements are highly susceptible to observation noise, asynchronicities, and other impediments that arise in the presence of uncertainty; as a result, algorithms requiring feedback of this type cannot be reliably implemented in real-world MIMO systems.

To relax this "perfect matrix feedback" requirement, [17] introduced a stochastic, first-order semidefinite optimization method based on *matrix exponential learning* (MXL). The MXL algorithm proceeds incrementally by combining stochastic gradient steps with a matrix exponential mapping that ensures feasibility of the users' signal covariance variables. In doing so, MXL guarantees fast convergence in cases where WF methods demonstrably fail: specifically, MXL achieves an ε-optimal state within $\mathcal{O}(1/\varepsilon^2)$ iterations, even in the presence of noise and uncertainty, in which case water-filling methods are known to produce suboptimal results [12, 13, 18].

On the negative side, MXL still requires (*a*) inverting a large matrix at the receiver; and (*b*) transmitting the resulting (dense) matrix to all connected users. In a MIMO array with $N = 128$ receive antennas, this means 65 kB of data per transmission frame, thus exceeding typical frame size limitations by a factor of 50× to 500× (depending on the specific standard) [19]. Coupled with the significant energy expenditures involved in matrix computations and the fact that entry-level antenna arrays may be ill-equipped for this purpose, the overhead of MXL quickly becomes prohibitive as MIMO systems "go large".

Contributions and related work: Our main objective in this paper is to lift the requirement that users have access to full matrix feedback at each transmission frame (e.g., perfect knowledge of their effective channel matrices or the system-wide signal-plus-noise covariance matrix). Our main tool to lift these feedback requirements is the introduction of a "zeroth-order" optimization framework in which gradients are estimated from observed throughput values using a technique known as *simultaneous perturbation stochastic approximation* (SPSA) [20, 21]. By integrating this SPSA technique in the chassis of the MXL method, we obtain a novel algorithm, which we call *gradient-free matrix exponential learning* (MXL0), and which we show converges to ε-optimality within $\mathcal{O}(1/\varepsilon^4)$ iterations (on average and with high probability).

On the positive side, this analysis shows that MXL0 is an asymptotically optimal algorithm (similarly to MXL, IWF and IW-MMSE) but *without* the full matrix feedback requirements of these methods. On the negative side, despite the vastly reduced feedback and overhead requirements of MXL0, the drop in convergence speed relative to the original MXL scheme is substantial and makes the algorithm ill-suited for practical systems. In fact, as we show via numerical experiments in realistic network conditions, MXL0 might take up to 10^5 iterations to achieve a relative optimality threshold of $\varepsilon = 10^{-1}$ (compared to between 10 and 100 iterations for MXL). This is caused by the very high variance of the SPSA estimator, which incurs a significant amount of state space exploration and leads to a dramatic drop in the algorithm's convergence speed.

To circumvent this obstacle, we introduce a variance reduction mechanism where information from previous transmit cycles is reused to improve the accuracy of the SPSA gradient estimator. We call the resulting algorithm *gradient-free MXL with callbacks* (MXL0$^+$), and we show that *it combines the best of both worlds:* it retains the fast $\mathcal{O}(1/\varepsilon^2)$ convergence rate of the standard MXL algorithm, despite the fact that it only requires a *single scalar* worth of feedback per iteration. In fact, in many instances, the reuse of past queries is so efficient that the gradient-free MXL0$^+$ algorithm ends up outperforming even MXL (which requires first-order gradient feedback).

With regard to feedback reduction, the work which is closest in spirit to our own is the very recent paper [22], where the authors seek to minimize the informational exchange of MXL methods applied to the maximization of transmit energy efficiency (as opposed to throughput). There, instead of requiring an $N \times N$ Hermitian matrix as feedback, each transmitter is assumed to receive a random selection of gradient components. This (batch) "coordinate descent" approach leads to a trade-off between signalling overhead and speed of convergence, but still relies on users having access to first-order gradient information. In contrast, we do not make any such assumptions and work *solely* with throughput observations; in this way, the communication overhead is reduced to a single scalar, while retaining the possibility of asynchronous, distributed updates.

Finally, from a beamforming perspective, the algebraic power method can also be used to iteratively approximate optimal beamformer/combiner pairs without prior knowledge of the channel matrix. However, this approach requires a stationary wireless background: in the presence of multiple users, user-to-user interference can render the estimation of individual channel matrices impossible. For this reason, we do not consider

such methods in the sequel; for an overview, see [23, 24].

***Notation**:* Throughout the sequel, we use bold symbols for matrices, saving the letters k, ℓ for user assignments and t, s for time indices, so that e.g., matrix $\mathbf{Q}_k$ relates to user k, $\mathbf{Q}_t$ to time t, and $\mathbf{Q}_{k,t}$ to user k at time t. The symbols $o(\cdot)$, $\mathcal{O}(\cdot)$, and $\Theta(\cdot)$ are taken as in the common Bachmann-Landau notation.

II. Problem Statement

In this section, we present two archetypal multi-user MIMO system models that are at the core of our considerations: a centralized sum-rate optimization problem, and an individual rate maximization game. In both cases, the optimization process is assumed to unfold in a distributed, online manner as follows:

1) At each transmission frame, every user in the network selects an *action* (an input signal covariance matrix).
2) This choice generates each user's *utility* (their sum- or individual rate, depending on the problem's specifics).
3) Based on the observed utilities, the users update their actions and the process repeats.

We stress here that *we do not assume* the existence of a centralized control hub with access to all the primitives defining the problem (individual channel matrices, input signal covariance matrices, etc.) and/or the capability of implementing an *offline* optimization algorithm to solve it. Instead, we focus on wireless networks with light-weight deployment and implementation characteristics, such as multi-user MIMO uplink networks in typical urban environments. In the downlink, the decision process regarding all transmission aspects (including the input signal covariance matrices) is inherently centralized as it takes places at the unique transmitter, which makes the broadcast setting a more resource-hungry choice compared to the uplink; nevertheless, the duality between the MAC and the broadcast channel (BC) [25] can be exploited to solve the analogous centralized problem in the downlink.

In terms of decoding, we consider two different schemes at the receiver: (*a*) *successive interference cancellation* (SIC), which is suitable for networks with centralized user admission and control protocols; and (*b*) *single user decoding* (SUD), which is suitable for more decentralized, ad hoc networks.

A. Centralized sum-rate maximization

Consider a Gaussian vector multiple access channel consisting of K users simultaneously transmitting to a wireless receiver equipped with N antennas. If the k-th transmitter is equipped with M_k antennas, we get the baseband signal model

$$\mathbf{y} = \sum\nolimits_{k=1}^{K} \mathbf{H}_k \mathbf{x}_k + \mathbf{z}, \tag{1}$$

where: (*a*) $\mathbf{x}_k \in \mathbb{C}^{M_k}$ denotes the signal transmitted by the k-th user; (*b*) $\mathbf{H}_k \in \mathbb{C}^{N \times M_k}$ is the corresponding channel matrix; (*c*) $\mathbf{y} \in \mathbb{C}^N$ is the aggregate signal reaching the receiver; and (*d*) $\mathbf{z} \in \mathbb{C}^N$ denotes the ambient noise in the channel, including thermal and environmental interference effects (and modeled for simplicity as a zero-mean, circulant Gaussian vector with identity covariance). In this general model, the transmit power of the k-th user is given by $p_k = \mathbb{E}[\mathbf{x}_k^\dagger \mathbf{x}_k]$. Then, letting P_k denote the maximum transmit power of user k, we also write

$$\mathbf{Q}_k = \mathbb{E}[\mathbf{x}_k \mathbf{x}_k^\dagger]/P_k \tag{2}$$

for the normalized signal (or input) covariance matrix of user k. By definition, $\mathbf{Q}_k$ is Hermitian and positive-semidefinite, which we denote by writing $\mathbf{Q}_k \in \mathrm{Herm}\,(M_k)$ and $\mathbf{Q}_k \succcurlyeq 0$ respectively.

Assuming successive interference cancellation (SIC) at the receiver, the users'achievable sum rate is given by the familiar expression

$$R(\mathbf{Q}) = \log\det \mathbf{W}, \tag{3}$$

where

$$\mathbf{W} \equiv \mathbf{W}(\mathbf{Q}) = \mathbf{I} + \sum\nolimits_{k=1}^{K} P_k \, \mathbf{H}_k \mathbf{Q}_k \mathbf{H}_k^\dagger \tag{4}$$

is the aggregate signal-plus-noise covariance matrix at the receiver, and $\mathbf{Q} \equiv (\mathbf{Q}_1, \ldots, \mathbf{Q}_K)$ denotes the users' aggregate signal covariance profile [26]. SIC decoding of this type has been exploited as a means to control the multi-user interference in the power-domain non-orthogonal multiple access (NOMA) technology [27], which provides a better spectrum utilization and spectral efficiency compared with traditional orthogonal schemes.

Since $R(\mathbf{Q})$ is increasing in each user's total transmit power $p_k = P_k \mathrm{tr}(\mathbf{Q}_k)$, the channel's throughput is maximized when the users individually saturate their power constraints, i.e., when $\mathrm{tr}(\mathbf{Q}_k) = 1$ for all $k = 1, \ldots, K$. In this way, we obtain the *power-constrained sum-rate optimization problem*

$$\begin{aligned} &\text{maximize} && R(\mathbf{Q}) \equiv R(\mathbf{Q}_1, \ldots, \mathbf{Q}_K) \\ &\text{subject to} && \mathbf{Q}_k \in \mathcal{Q}_k \text{ for all } k = 1, \ldots, K, \end{aligned} \tag{Opt}$$

where each user's feasible power region $\mathcal{Q}_k$ is given by

$$\mathcal{Q}_k = \{\mathbf{Q}_k \in \mathrm{Herm}\,(M_k) : \mathrm{tr}(\mathbf{Q}_k) = 1, \mathbf{Q}_k \succcurlyeq 0\}. \tag{5}$$

By definition, each $\mathcal{Q}_k$ is a spectrahedron of (real) dimension $d_k = M_k^2 - 1$, so the problem's dimensionality is $\sum_k d_k = \mathcal{O}(\sum_k M_k^2)$. To avoid trivialities, we will assume in what follows that each transmitter possesses at least two antennas, so $d_k > 0$ for all $k = 1, \ldots, K$. Also, to further streamline our discussion, we will state our results in terms of the maximum number $M = \max_k M_k$ of antennas per transmitter—or, equivalently, in terms of the larger dimension $d = M^2 - 1$.[2]

B. Distributed individual rate maximization

Moving beyond the sum-rate maximization problem above, if messages are decoded using single user decoding at the receiver (i.e., interference by all other users is treated as additive colored noise), each user's *individual* rate will be

$$R_k(\mathbf{Q}_k; \mathbf{Q}_{-k}) = R(\mathbf{Q}_1, \ldots, \mathbf{Q}_K) - R(\mathbf{Q}_1, \ldots, 0, \ldots, \mathbf{Q}_K), \tag{6}$$

where $(\mathbf{Q}_k; \mathbf{Q}_{-k})$ is shorthand for the covariance profile $(\mathbf{Q}_1, \ldots, \mathbf{Q}_k, \ldots, \mathbf{Q}_K)$. In turn, this leads to the *individual rate maximization* game

$$\begin{aligned} &\text{maximize} && R_k(\mathbf{Q}_k; \mathbf{Q}_{-k}) \\ &\text{subject to} && \mathbf{Q}_k \in \mathcal{Q}_k \end{aligned} \tag{Opt$_k$}$$

[2]The statement of our results can be fine-tuned at the cost of introducing further notation for other aggregate statistics of the number of antennas per transmitter (such as the arithmetic or geometric mean of M_k). The resulting expressions are fairly cumbersome, so we do not report them here.

to be solved unilaterally by each user $k = 1, \dots, K$.

Given that $R(\mathbf{Q})$ is concave in $\mathbf{Q}$ and $R_k(\mathbf{Q}_k; \mathbf{Q}_{-k})$ is concave in $\mathbf{Q}_k$, it follows that the decentralized problem (Opt_k) defines a concave potential game whose Nash equilibria coincide with the solutions of (Opt) [17, 28, 29]. In view of this, the gradient-free optimization framework and algorithms derived in this paper and designed to solve the centralized sum-rate optimization (Opt) will also solve the game (Opt_k); conversely, (Opt) is amenable to a distributed approach where it is treated as the aggregation of the unilateral sub-problems (Opt_k), to be solved in parallel by the network's users. We revisit this distributed approach in Section V.

C. Water-filling and matrix exponential learning

A basic online solution method for (Opt) is the water-filling (WF) algorithm [7, 8, 18] and its variants—iterative or simultaneous [12, 13, 30]. In WF schemes, transmitters are tacitly assumed to have full knowledge of their channel matrices $\mathbf{H}_k$ as well as the multi-user interference-plus-noise (MUI) covariance matrix

$$\mathbf{W}_k = \mathbf{I} + \sum\nolimits_{\ell \neq k} P_\ell \mathbf{H}_\ell \mathbf{Q}_\ell \mathbf{H}_\ell^\dagger. \tag{7}$$

These matrices are then used to "water-fill" the users' *effective channel matrices*

$$\tilde{\mathbf{H}}_k = \mathbf{W}_k^{-1/2} \mathbf{H}_k \tag{8}$$

either iteratively (i.e., in a round-robin fashion), or simultaneously (all transmitters at the same time); the corresponding implementations are called iterative water-filling (IWF) and simultaneous water-filling (SWF) respectively.

We stress here that the users' effective channel matrices may change over time, even when the *actual* channel matrix $\mathbf{H}_k$ is static: this is because $\tilde{\mathbf{H}}_k$ depends on the transmission characteristics of *all* other users in the network (via the MUI matrix $\mathbf{W}_k$), and these typically evolve over time according to each user's optimization policy.

In this context, IWF converges *always* (but slowly if the number of users is large), whereas SWF *may fail* to converge altogether [13, 31]. In addition, as we discussed in the introduction, water-filling is highly susceptible to observation noise, asynchronicities, and other impediments that arise in real-world systems, so the solution of (Opt) in the presence of uncertainty requires a different approach (see also the numerical experiments presented in Section VI).

These limitations are overcome by the *matrix exponential learning* (MXL) algorithm [17, 32], which will serve both as a reference and an entry point for our analysis. Heuristically, MXL proceeds by aggregating incremental gradient steps (possibly evaluated with imperfect channel state and MUI estimations), and then using a suitable matrix exponential mapping to convert these steps into a positive-semidefinite matrix that meets the transmit power constraints of (Opt) and/or (Opt_k).

More formally, let

$$\nabla_k R(\mathbf{Q}) = P_k \mathbf{H}_k^\dagger \left[\mathbf{I} + \sum\nolimits_{\ell=1}^{K} P_\ell \, \mathbf{H}_\ell \mathbf{Q}_\ell \mathbf{H}_\ell^\dagger \right]^{-1} \mathbf{H}_k. \tag{9}$$

denote the individual gradient of R (or R_k) relative to the signal covariance matrix of the k-th user, and let

$$\mathcal{Y}_k = \{ \mathbf{Y}_k \in \text{Herm}\,(M_k) : \text{tr}(\mathbf{Y}_k) = 0 \} \tag{10}$$

denote the subspace tangent to $\mathcal{Q}$. Then, given an initialization $\mathbf{Y}_1 \in \mathcal{Y} \equiv \prod_k \mathcal{Y}_k$, the MXL algorithm is defined via the basic recursion

$$\begin{aligned} \mathbf{Q}_t &= \mathbf{\Lambda}(\mathbf{Y}_t), \\ \mathbf{Y}_{t+1} &= \mathbf{Y}_t + \gamma_t \mathbf{V}_t, \end{aligned} \tag{MXL}$$

where:

i) $\mathbf{Q}_t$ denotes the users' input signal covariance profile at the t-th iteration of the algorithm ($t = 1, 2, \dots$).

ii) $\mathbf{V}_t = (\mathbf{V}_{1,t}, \dots, \mathbf{V}_{K,t})$ is an estimate of the tangent component of the gradient ∇R relative to $\mathcal{Q}$.[3]

iii) $\gamma_t > 0$ is a non-increasing sequence of step-sizes whose role is examined in detail below.

iv) $\mathbf{Y}_t$ is an auxiliary matrix that aggregates gradient steps.

v) $\mathbf{\Lambda}(\mathbf{Y}) = (\mathbf{\Lambda}_1(\mathbf{Y}_1), \dots, \mathbf{\Lambda}_K(\mathbf{Y}_K))$ denotes the matrix exponential mapping given in (block) components by

$$\mathbf{\Lambda}_k(\mathbf{Y}_k) = \frac{\exp(\mathbf{Y}_k)}{\text{tr}(\exp(\mathbf{Y}_k))}. \tag{11}$$

The intuition behind (MXL) is that the exponential mapping assigns more power to the spatial directions that are aligned to the objective's gradient (as estimated via $\mathbf{V}_t$). In fact, the MXL algorithm can be explained as a matrix-valued instance of Nesterov's dual averaging method [33]; the key innovation of MXL is the matrix exponentiation step which lifts the need to do a costly projection on the users' feasible region (a trace-constrained spectrahedron). The output of each iteration of the algorithm is a positive-semidefinite matrix with unit trace, so the problem's constraints are automatically satisfied. We defer the details of this derivation to Appendix A.

As was shown in [17], the MXL algorithm achieves an ε-optimal signal covariance profile within $\mathcal{O}(1/\varepsilon^2)$ iterations. However, to do so, the algorithm still requires access to noisy observations of the gradient matrices (9). Typically, this involves inverting a (dense) $N \times N$ Hermitian matrix at a central hub and subsequently transmitting the result to the network's users, so the algorithm's computation and communication overhead is considerable (see Table I). On that account, our main focus in the sequel will be to lift the assumption that the network's users have access to the gradient matrices (9), all the while maintaining the $\mathcal{O}(1/\varepsilon^2)$ convergence speed of (MXL).

D. Technical preliminaries and notation

For the analysis to come, it will be convenient to introduce the following constants. First, we will write $\mathcal{Q} = \prod_k \mathcal{Q}_k$ for the feasible region of (Opt), and we will denote by L the Lipschitz constant of R over $\mathcal{Q}$ relative to the nuclear norm; specifically, this means that:

$$|R(\mathbf{Q}) - R(\mathbf{Q}')| \leq L \|\mathbf{Q} - \mathbf{Q}'\| \quad \text{for all } \mathbf{Q}, \mathbf{Q}' \in \mathcal{Q}. \tag{12}$$

Moreover, we will also write $\lambda_{k\ell}$ for the user-specific Lipschitz constants of $\nabla_k R$, understood in the following sense:

$$\|\nabla_k R(\mathbf{Q}_\ell; \mathbf{Q}_{-\ell}) - \nabla_k R(\mathbf{Q}'_\ell; \mathbf{Q}_{-\ell})\|_* \leq \lambda_{k\ell} \|\mathbf{Q}_\ell - \mathbf{Q}'_\ell\|_2, \tag{13}$$

[3] More precisely, (MXL) only requires estimates of $\nabla R : \mathcal{Q} \mapsto \mathcal{Y}$, which here denotes the tangent component of the gradient ∇R relative to $\mathcal{Q}$, given by $\nabla R = (\nabla_1 R, \dots, \nabla_K R)$ where $\nabla_k R = \nabla_k R - \text{tr}(\nabla_k R)\, \mathbf{I}$. All technical details in regards to (MXL) are deferred to the appendix.

for all $\mathbf{Q}_\ell, \mathbf{Q}'_\ell \in \mathcal{Q}_\ell$, $\mathbf{Q}_{-\ell} \in \mathcal{Q}_{-\ell} \equiv \prod_{j\neq\ell} \mathcal{Q}_j$, and all $k, \ell = 1, \dots, K$. We also let $\lambda_k = (1/K) \sum_{\ell=1}^K \lambda_{k\ell}$ denote the "averaged" Lipschitz constant of user k, and we write $\lambda = (1/K) \sum_{k=1}^K \lambda_k$ for the overall "mean" Lipschitz constant. For a detailed discussion of the nuclear norm $\|\cdot\|$ and its dual $\|\cdot\|_*$, we refer the reader to Appendix A.

III. MXL without Gradient Information

As we noted above, the existing implementations of MXL invariably rely on the availability of gradient feedback—full [32], noisy [17], or partial [22]. Our aim in this section is to show that this requirement can be obviated by means of a (possibly biased) gradient estimator, which only requires observations of a *single* scalar—the users' achieved throughput. Our approach builds on the method of *simultaneous perturbation stochastic approximation* (SPSA), a gradient estimation procedure which has been studied extensively in the context of large-scale, derivative-free optimization [20, 21], and which we discuss in detail below.

A. Gradient estimation: intuition and formal construction

We start by providing some intution behind the SPSA method. For this, consider the scalar case and a simple differentiable function $f : \mathbb{R} \mapsto \mathbb{R}$. Then, by definition, the derivative of f at any point x satisfies

$$f'(x) = \frac{f(x+\delta) - f(x-\delta)}{2\delta} + o(\delta). \tag{14}$$

Therefore, if $\delta > 0$ is small enough, an estimate for $f'(x)$ can be obtained from two queries of the value of f at the neighboring points $x - \delta$ and $x + \delta$ as follows:

$$\hat{v}(x) = \frac{f(x+\delta) - f(x-\delta)}{2\delta}. \tag{15}$$

Thus, if f' is λ-Lipschitz continuous on the search domain, it is easy to see that the error of the estimator $\hat{v}(x)$ is uniformly bounded as $|\hat{v}(x) - f'(x)| \leq \lambda\,\delta/2$, i.e., the estimator (15) is accurate up to $\mathcal{O}(\delta)$.

Taking this idea further, it is possible to estimate $f'(x)$ using only a *single* function query at either of the test points $x - \delta$, or $x + \delta$, chosen uniformly at random. To carry this out, let z be a random variable taking the value -1 or $+1$ with equal probability $1/2$, and define the *one-shot* SPSA estimator

$$v(x) = \frac{f(x+\delta z)}{\delta} z. \tag{16}$$

Then, a straightforward calculation gives $\mathbb{E}[v(x)] = \hat{v}(x)$, i.e., v is a stochastic estimator of f' with accuracy

$$|\mathbb{E}[v(x) - f'(x)]| = |\hat{v}(x) - f'(x)| \leq \lambda\,\delta/2 = \mathcal{O}(\delta). \tag{17}$$

The SPSA approach described above can be applied to our MIMO setting as follows. First, each user k draws, randomly and independently, a matrix $\mathbf{Z}_k$ from the unit sphere[4]

$$\mathbb{S}^{d_k - 1} = \{\mathbf{Z}_k \in \mathcal{Y}_k : \|\mathbf{Z}_k\|_2 = 1\}. \tag{18}$$

[4]Note that the dimension of $\mathbb{S}^{d_k-1}$ as a manifold is $d_k - 1$, i.e., one lower than that of the feasible region $\mathcal{Q}_k$; this is due to the unit norm constraint $\|\mathbf{Z}_k\|_2 = 1$.

Then, translating (16) to the distributed, Hermitian setting of Section II yields, for all $k = 1, \dots, K$, the gradient estimator

$$\mathbf{V}_k(\mathbf{Q}) = \frac{d_k}{\delta} R(\mathbf{Q} + \delta\mathbf{Z})\,\mathbf{Z}_k, \tag{19}$$

where $\mathbf{Z} = (\mathbf{Z}_1, \dots, \mathbf{Z}_K)$ collects the random shifts of all users.

Remark 1. The factor $d_k = M_k^2 - 1$ in (19) has a geometric interpretation as the ratio between the volumes of the sphere $\mathbb{S}^{d_k-1}$ (where $\mathbf{Z}_k$ is drawn from) and the containing d_k-dimensional ball $\mathbb{B}^{d_k} = \{\mathbf{Z}_k \in \mathcal{Y}_k : \|\mathbf{Z}_k\|_2 \leq 1\}$. Its presence is due to Stokes' theorem, as detailed in Lemma B.1.

A further complication that arises in our constrained setting is that the query point $\mathbf{Q}+\delta\mathbf{Z}$ in (19) may lie outside the feasible set $\mathcal{Q}$ if $\mathbf{Q}$ is too close to the boundary of $\mathcal{Q}$. To avoid such an occurrence, we introduce below a "safety net" mechanism which systematically carries back the pivot points $\mathbf{Q}_k$ towards the "prox-center" $\mathbf{C}_k = \mathbf{I}_{M_k}/M_k$ of $\mathcal{Q}_k$ before applying the random shift $\mathbf{Z}_k$. Specifically, taking $r_k > 0$ sufficiently small so that the Frobenius ball centered at $\mathbf{C}_k$ lies entirely in $\mathcal{Q}_k$, we consider the homothetic adjustment

$$\hat{\mathbf{Q}}_k = \mathbf{Q}_k + \frac{\delta}{r_k}(\mathbf{C}_k - \mathbf{Q}_k) + \delta\mathbf{Z}_k. \tag{20}$$

By an elementary geometric argument, it suffices to take

$$r_k = 1/\sqrt{M_k(M_k - 1)}. \tag{21}$$

With this choice of r_k, it is easy to show that, for $\delta < r_k$, the adjusted query point $\hat{\mathbf{Q}}_k$ lies in $\mathcal{Q}_k$ for all $k = 1, \dots, K$. On that account, we redefine the SPSA estimator for (Opt) as

$$\mathbf{V}_k(\mathbf{Q}) = \frac{d_k}{\delta} R(\hat{\mathbf{Q}})\,\mathbf{Z}_k, \tag{SPSA}$$

where, in obvious notation, we set $\hat{\mathbf{Q}} = (\hat{\mathbf{Q}}_1, \dots, \hat{\mathbf{Q}}_K)$. The distinguishing feature of (SPSA) is that it is well-posed: *any* query point $\hat{\mathbf{Q}}$ is feasible under (SPSA). Thus, extending the one-dimensional analysis in the beginning of this section, Lemma B.1 claims that the accuracy of the estimator (SPSA) is uniformly bounded as $\|\mathbb{E}[\mathbf{V}_k(\mathbf{Q}) - \nabla R(\mathbf{Q})]\|_* = \mathcal{O}(\delta)$. In the rest of this section, we exploit this property to derive and analyze a first *gradient-free* variant of (MXL).

B. A gradient-free matrix exponential learning scheme

To integrate the gradient estimator (SPSA) in the chassis of (MXL), we will use a (non-increasing) query radius sequence δ_t satisfying the basic feasibility condition:

$$\delta_t < \min_k r_k = 1/\sqrt{M(M-1)} \quad \text{for all } t \geq 1. \tag{H0}$$

Then, under (MXL), the task of user k at the t-th stage of the algorithm will be given by the following sequence of events:

1) Draw a random direction $\mathbf{Z}_{k,t} \in \mathbb{S}^{d_k-1}$.
2) Transmit with the covariance matrix $\hat{\mathbf{Q}}_{k,t}$ given by (20).
3) Get the achieved throughput $\hat{R}_t = R(\hat{\mathbf{Q}}_t)$.
4) Construct the gradient estimate $\mathbf{V}_{k,t}$ given by (SPSA).
5) Update $\mathbf{Y}_{k,t}$ and $\mathbf{Q}_{k,t}$ in accordance with (MXL).

The resulting algorithm will be referred to as *gradient-free matrix exponential learning* (MXL0); for a pseudocode implementation, see Alg. 1 above.

Algorithm 1: Gradient-free matrix exponential learning (MXL0)

Parameters: γ_t, δ_t
Initialization: $t \leftarrow 1$, $\mathbf{Y} \leftarrow 0$;
$\forall k\colon \mathbf{Q}_k \leftarrow (P_k/M_k)\,\mathbf{I}_k$
Repeat
  For $k \in \{1,\dots,K\}$ **do** $\texttt{MXL0}_k(\gamma_t,\delta_t)$ **in parallel**
  $t \leftarrow t+1$

Routine $\texttt{MXL0}_k(\gamma,\delta)$ **:**
  Sample $\mathbf{Z}_k$ **uniformly over** $\mathbb{S}^{d_k-1}$
  Transmit with $\hat{\mathbf{Q}}_k \leftarrow \mathbf{Q}_k + \frac{\delta}{r_k}(\mathbf{C}_k - \mathbf{Q}_k) + \delta\mathbf{Z}_k$
  Get $\hat{R} \leftarrow R(\hat{\mathbf{Q}})$
  Set $\mathbf{V}_k \leftarrow \frac{d_k}{\delta}\hat{R}\,\mathbf{Z}_k$
  Set $\mathbf{Y}_k \leftarrow \mathbf{Y}_k + \gamma\mathbf{V}_k$
  Set $\mathbf{Q}_k \leftarrow \mathbf{\Lambda}_k(\mathbf{Y}_k)$

Our first convergence result for MXL0 is as follows:

Theorem 1 (Convergence of MXL0)**.** *Suppose that MXL0 (Alg. 1) is run with non-increasing step-size and query-radius policies satisfying* (H0) *and*

$$\text{(a) } \textstyle\sum_t \gamma_t = \infty, \quad \text{(b) } \sum_t \gamma_t\delta_t < \infty, \quad \text{(c) } \sum_t \gamma_t^2/\delta_t^2 < \infty. \tag{22}$$

Then, with probability 1*, the sequence of the users' transmit covariance matrices* $\hat{\mathbf{Q}}_t$ *converges to the solution set of* (Opt)*.*

Theorem 1 provides a strong asymptotic convergence result, but it does not give any indication of the algorithm's convergence speed. To fill this gap, our next result focuses on the algorithm's value convergence rate relative to the maximum achievable transmission rate $R^* = \max R$ of (Opt).

Theorem 2 (Convergence rate of MXL0)**.** *Suppose that MXL0 (Alg. 1) is run for T iterations with constant step-size and query radius parameters of the form $\gamma_t = \gamma/T^{3/4}$ and $\delta_t = \delta/T^{1/4}$, $\delta < 1/\sqrt{M(M-1)}$. Then, the algorithm's ergodic average $\bar{\mathbf{Q}}_T = (1/T)\sum_{t=1}^T \mathbf{Q}_t$ enjoys the bounds:*

(a) In expectation,

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] \leq \frac{A(\gamma,\delta)}{T^{1/4}} = \mathcal{O}\left(T^{-1/4}\right), \tag{23}$$

where $A(\gamma,\delta) = (K/\gamma)\log M + 4K^2\lambda\delta + 2^{1-2K}(R^*d)^2K\gamma/\kappa\delta^2$.

(b) In probability, for any small enough tolerance $\varepsilon > 0$,

$$\mathbb{P}\left(R^* - R(\bar{\mathbf{Q}}_T) \geq \frac{A(\gamma,\delta)}{T^{1/4}} + \varepsilon\right) \leq \exp\left(-\frac{2^{2K-5}\delta^2\varepsilon^2T^{1/2}}{(R^*Kd)^2}\right). \tag{24}$$

In words, Theorem 2 shows that Alg. 1 converges at a rate of $\mathcal{O}(T^{-1/4})$ on average, and the probability of deviating by more than ε from this rate is exponentially small in ε and T. Compared to (MXL), this indicates an increase in the number of iterations required to achieve ε-optimality from $\mathcal{O}(1/\varepsilon^2)$ to $\mathcal{O}(1/\varepsilon^4)$. As we illustrate in detail in Section VI, this performance drop is quite significant and makes MXL0 prohibitively slow in practice. The rest of our paper is devoted precisely to bridging this vital performance gap.

IV. Accelerated MXL without Gradient Information

Going back to the heuristic discussion of MXL0 in the previous section, we see that the one-shot estimator v is bounded as $|v| \leq \sup|f|/\delta = \mathcal{O}(1/\delta)$. This unveils a significant trade-off between the $\mathcal{O}(\delta)$ bias of the estimator and its $\mathcal{O}(1/\delta)$ deviation from the true derivative: the more accurate v becomes (smaller bias), the less precise it will be (higher variance). In the context of iterative optimization algorithms, this *bias–variance dilemma* induces strict restrictions on the design of the query-radius and step-size policies, with deleterious effects on the algorithm's convergence rate (cf. Sections III and VI). Motivated by this drawback of the SPSA approach, we proceed in the sequel to design a gradient estimator which requires a single function query per iteration, whilst at the same time enjoying a uniform bound on the norms of the estimates.

A. SPSA with callbacks

To proceed with our construction, let z take the value -1 or $+1$ with equal probability, and consider the estimator

$$v_\rho(x) = \frac{f(x+\delta z) - \rho}{\delta} z. \tag{25}$$

The offset value ρ is decided *a priori*, independently of the random variable z, so that $\mathbb{E}[\rho z] = \rho\mathbb{E}[z] = 0$. In turn, this implies that $\mathbb{E}[v_\rho(x)] = \hat{v}(x)$, and hence:

$$|\mathbb{E}[v_\rho(x) - f'(x)]| \leq \lambda\delta/2 \tag{26}$$

i.e., the accuracy (bias) of $v_\rho(x)$ is again $\mathcal{O}(\delta)$.

The novelty of (25) is as follows: if we take $\rho = f(x)$, then $|v_\rho(x)| = (1/\delta)|f(x+\delta z) - f(x)| \leq L$ where L denotes the Lipschitz constant of f, so the choice $\rho = f(x)$ would be ideally suited for our purposes; however, taking $\rho = f(x)$ would also involve an additional function query. To circumvent this, we will instead approximate $f(x)$ with the closest available surrogate, namely the function value observed at the previous iteration of the process.

To make this precise in our MIMO context, we will consider the *enhanced* SPSA estimator

$$\mathbf{V}_{k,t} = \frac{d_k}{\delta_t}\left[R(\hat{\mathbf{Q}}_t) - R(\hat{\mathbf{Q}}_{t-1})\right]\mathbf{Z}_{k,t}, \tag{SPSA+}$$

where:

1) δ_t is the given query radius at time t.
2) $\mathbf{Z}_{k,t}$ is drawn randomly from the sphere $\mathbb{S}^{d_k-1}$
3) $\hat{\mathbf{Q}}_t$ is the transmit covariance matrix defined along (20).

Then, integrating (SPSA+) in the chassis of MXL, we obtain a similarly enhanced version of MXL0, which we call *gradient-free MXL with callbacks* (MXL0$^+$). For concreteness, we present a pseudocode implementation of the resulting method in Alg. 2.

In terms of parameter values, MXL0$^+$ supports a broad class of policies satisfying the so-called Robbins–Monro conditions:

$$\text{(a) } \textstyle\sum_t \gamma_t = \infty, \qquad \text{(b) } \sum_t \gamma_t^2 < \infty. \tag{H1}$$

Algorithm 2: Gradient-free MXL with callbacks (MXL0$^+$)

Parameters: γ_t, δ_t
Initialization: $t \leftarrow 1$, $\mathbf{Y} \leftarrow 0$;
$\forall k\colon \mathbf{Q}_k \leftarrow (P_k/M_k)\,\mathbf{I}_k$, $\rho_k \leftarrow R(\mathbf{Q})$

Repeat
  For $k \in \{1,\dots,K\}$ **do** $\texttt{MXL0}_k^+(\gamma_t,\delta_t)$ **in parallel**
  $t \leftarrow t+1$

Routine $\texttt{MXL0}_k^+(\gamma,\delta)$:
  Sample $\mathbf{Z}_k$ **uniformly over** $\mathbb{S}^{d_k-1}$
  Transmit with $\hat{\mathbf{Q}}_k \leftarrow \mathbf{Q}_k + \frac{\delta}{r_k}(\mathbf{C}_k - \mathbf{Q}_k) + \delta\mathbf{Z}_k$
  Get $\hat{R} \leftarrow R(\hat{\mathbf{Q}})$
  Set $\mathbf{V}_k \leftarrow \frac{d_k}{\delta}(\hat{R} - \rho_k)\,\mathbf{Z}_k$
  Set $\rho_k \leftarrow \hat{R}$
  Set $\mathbf{Y}_k \leftarrow \mathbf{Y}_k + \gamma\mathbf{V}_k$
  Set $\mathbf{Q}_k \leftarrow \mathbf{\Lambda}_k(\mathbf{Y}_k)$

In addition, MXL0$^+$ also requires the following precautions regarding the allowable step-size and query-radius sequences:

$$\textstyle\sum_t \gamma_t\delta_t < \infty, \tag{H2}$$

$$\sup_t \gamma_t/\delta_{t+1} < 2/(dLK), \tag{H3}$$

$$\sup_t \delta_t/\delta_{t+1} < \infty, \tag{H4}$$

Of the above conditions, (H3)–(H4) guarantee the uniform boundedness of the gradient estimator, while (H2) is an additional condition needed for convergence of the algorithm.

In practice, these conditions are easy to verify when $\gamma_t = \gamma/t^\alpha$ and $\delta_t = \delta/t^\beta$ for some $\alpha,\beta > 0$. In this case, the conditions (H0)–(H4) reduce to:

$$(dLK/2)\,\gamma < \delta < 1/\sqrt{M(M-1)}, \tag{Ha}$$

$$0 \le \beta \le \alpha \le 1 \quad \text{and} \quad \alpha+\beta > 1, \tag{Hb}$$

With all this in hand, we are finally in a position to state our main convergence results for the MXL0$^+$ algorithm. We begin by establishing the algorithm's almost sure convergence:

Theorem 3 (Convergence of MXL0$^+$)**.** *Suppose that MXL0$^+$ (Alg. 2) is run with step-size and query-radius policies satisfying* (H0)–(H4)*. Then, with probability* 1*, the sequence of the users' transmit covariance matrices $\hat{\mathbf{Q}}_t$ converges to the solution set of* (Opt)*.*

As in the case of Theorem 1, Theorem 3 provides a strong asymptotic convergence result, but it leaves open the crucial question of the algorithm's convergence speed. Our next result justifies the introduction of (SPSA+) and shows that Alg. 2 achieves the best of both worlds: one-shot throughput measurements with an $\tilde{\mathcal{O}}(1/\sqrt{T})$ convergence rate.

Theorem 4 (Convergence rate of MXL0$^+$)**.** *Suppose that MXL0$^+$ (Alg. 2) is run for T iterations. We then have:*

1) If $\gamma_t = \gamma/\sqrt{t}$ and $\delta_t = \delta/\sqrt{t}$ with γ and δ satisfying (Ha)*:*

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] = \mathcal{O}\left(\frac{\log T}{\sqrt{T}}\right). \tag{27}$$

2) If $\gamma_t = \gamma/\sqrt{T}$ and $\delta_t = \delta/\sqrt{T}$ with γ and δ satisfying (Ha)*:*

TABLE II: Parameters of MXL0$^+$ for Corollary 1

a)	$\gamma = \dfrac{\sqrt{\log M/(dLK^2)}}{\sqrt{\lambda}+\sqrt{2dLK}}$; $\quad \delta = \dfrac{\sqrt{(dL/\lambda)\log M}}{2}$; $\quad T \ge \dfrac{LM^4\log M}{4\lambda}$
b)	$\gamma = \frac{\phi(\alpha)}{\sqrt{L}}\Big[\sqrt{2L}d\,\phi(\alpha) + \sqrt{\sqrt{2/\log(1/\alpha)}\,\lambda}\Big]^{-1}\frac{\sqrt{\log M}}{K\sqrt{d}}$; $\delta = \frac{\phi(\alpha)}{2}\sqrt{L/\lambda}\sqrt{\log(1/\alpha)/2}\,d\log M$; $T = 4\phi^4(\alpha)L^2\Big[1+\frac{1}{\phi(\alpha)d}\sqrt{2\lambda/L\sqrt{2/\log(1/\alpha)}}\Big]^2\Big(\frac{\log(1/\alpha)K^4d^3\log M}{\varepsilon^2}\Big)$; with $\phi(\alpha) = \Big[1/\sqrt{\log(1/\alpha)} + 4/\sqrt{\log(M)}\Big]^{1/2}$

a) In expectation,

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] \le \frac{B(\gamma,\delta)}{\sqrt{T}} = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right), \tag{28}$$

where $B(\gamma,\delta) = (K/\gamma)\log M + 4K^2\lambda\delta + \frac{8Kd\gamma}{[2/(dLK)-\gamma/\delta]^2}$.

b) In probability, for any small enough tolerance $\varepsilon > 0$,

$$\mathbb{P}\left(R^* - R(\bar{\mathbf{Q}}_T) \ge \frac{B(\gamma,\delta)}{\sqrt{T}} + \varepsilon\right) \le \exp\left(-\frac{\varepsilon^2 T}{C(\gamma,\delta)}\right), \tag{29}$$

where $C(\gamma,\delta) = 2^9dK^2/[2/(dLK) - \gamma/\delta]^2$.

Importantly, Theorem 4 shows that MXL0$^+$ recovers the $\mathcal{O}(1/\sqrt{T})$ convergence rate of MXL with *full* gradient information, even though the network's users are no longer assumed to have *any* access to a gradient oracle. In fact, the guarantees of Theorem 4 can be optimized further by finetuning the choice of γ and δ; doing just that (and referring to Appendix D for the details), we have:

Corollary 1. *Suppose that MXL0$^+$ is run with $\gamma_t = \gamma/\sqrt{T}$, $\delta_t = \delta/\sqrt{T}$, and T, γ, δ as in Table II. Then:*

a) In expectation, we have:

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] \le 2L\left(1 + \frac{2^{3/4}\sqrt{\lambda/L}}{d}\right)\sqrt{\frac{K^4M^6\log M}{T}}. \tag{30}$$

b) In probability, given a small enough tolerance $\varepsilon > 0$ and a confidence level $1-\alpha \in (0,1)$, we have:

$$\mathbb{P}(R^* - R(\bar{\mathbf{Q}}_{\lceil T\rceil}) \le \varepsilon) \ge 1-\alpha. \tag{31}$$

An important feature of the convergence rate guarantee (31) is that it does not depend on the number of antennas N at the receiver. As such, Alg. 2 exhibits a *scale-free* behavior relative to N, which makes it particularly appealing for distributed massive-MIMO systems. In the next section, we further relax the requirement that all users update their transmit covariance matrices in a synchronous manner, and we derive a fully distributed version of the MXL0$^+$ algorithm.

V. Distributed Implementation

In this section, we propose a distributed variant of the MXL0$^+$ method which can account for randomized and asynchronous user decisions (independent or in alternance with other users). Specifically, we now assume that, at each stage of the process, only a random subset of users perform an update of their individual covariances matrices, while the remaining users maintain the same covariance matrix, without updating.

Algorithm 3: The asynchronous MXL0$^+$ (AMXL0$^+$) method

Parameters: Π, γ_t, δ_t
Initialization: $t \leftarrow 1$, $\mathbf{Y} \leftarrow 0$;
$\forall k\colon \mathbf{Q}_k \leftarrow (P_k/M_k)\,\mathbf{I}_k$, $\rho_k \leftarrow R(\mathbf{Q})$
Repeat
  Draw set of active users U **according to** Π
  For $k \in \{1,\dots,K\}$ **do in parallel**
    If $k \in U$ **then** $\mathtt{MXL0}_k^+(\gamma_t,\delta_t)$ **else** $\mathtt{Pass}_k$
  $t \leftarrow t+1$

Routine $\mathtt{Pass}_k$ **:**
  Transmit with $\mathbf{Q}_k$
  Get $\rho_k \leftarrow R(\mathbf{Q})$

To state this formally, suppose that a random subset of users $U_t \subseteq \mathcal{K} \equiv \{1,\dots,K\}$ is drawn at stage t following an underlying probability law $\Pi \equiv (\Pi_U)_{U\subseteq\mathcal{K}}$ (i.e., $U \subseteq \mathcal{K}$ is drawn with probability Π_U). From the distributed perspective of individual users, we write $\pi_k = \sum_{U\ni k}\Pi_U$ to denote the *marginal probability* that user k updates their covariance at any stage t; as such, the participation of all users is enforced by imposing the condition $\pi_k > 0$. We thus obtain the asynchronous MXL0$^+$ scheme:

$$\begin{aligned}\mathbf{Q}_t &= \mathbf{\Lambda}(\mathbf{Y}_t),\\ \mathbf{Y}_{t+1} &= \mathbf{Y}_t + \gamma_t\,\hat{\mathbf{V}}_t,\end{aligned} \qquad \text{(AMXL0}^+\text{)}$$

where $\hat{\mathbf{V}}_{k,t} = \mathbf{V}_{k,t}$ if $k \in U_t$, and $\hat{\mathbf{V}}_{k,t} = 0$ otherwise. For a pseudocode implementation, see also Alg. 3 above.

As we show below, AMXL0$^+$ recovers the $\mathcal{O}(1/\sqrt{T})$ convergence rate of MXL0$^+$, despite being distributed across users:

Theorem 5 (Convergence rate of AMXL0$^+$)**.** *Suppose that AMXL0$^+$ (Alg. 3) is run for T iterations. We then have:*

1) If $\gamma_t = \gamma/\sqrt{t}$ and $\delta_t = \delta/\sqrt{t}$ with γ and δ satisfying (Ha)*:*

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] = \mathcal{O}\left(\frac{\log T}{\sqrt{T}}\right). \tag{32}$$

2) If $\gamma_t = \gamma/\sqrt{T}$ and $\delta_t = \delta/\sqrt{T}$ with γ and δ satisfying (Ha)*:*

a) In expectation,

$$\mathbb{E}[R^* - R(\bar{\mathbf{Q}}_T)] \le \frac{B_\pi(\gamma,\delta)}{\sqrt{T}} = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right), \tag{33}$$

where $B_\pi(\gamma,\delta) = \sum_{k=1}^K \frac{\log M_k}{\pi_k\gamma} + 4K^2\lambda\delta + \frac{8Kd\gamma}{[2/(dLK)-\gamma/\delta]^2}$.

b) In probability, for any small enough tolerance $\varepsilon > 0$,

$$\mathbb{P}\left(R^* - R(\bar{\mathbf{Q}}_T) \ge \frac{B_\pi(\gamma,\delta)}{\sqrt{T}} + \varepsilon\right) \le \exp\left(-\frac{\varepsilon^2 T}{C_\pi(\gamma,\delta)}\right), \tag{34}$$

where $C_\pi(\gamma,\delta) = \left[1 + \frac{\upsilon_\pi}{2} + \frac{\upsilon_\pi d^{3/2}LK\gamma\delta}{2\delta - dLK\gamma}\right]^2 C(\gamma,\delta)$*, with* $\upsilon_\pi = K^{-1}\sum_{k=1}^K \max(1,\pi_k^{-1}-1)$ *and $C(\gamma,\delta)$ as in Theorem 4.*

Note here that the quantity $B_\pi(\gamma,\delta)$ above only differs from its counterpart $B(\gamma,\delta)$ of Theorem 4 in the first term, which measures the cost of asynchronicity in terms of expected convergence. A similar increase in the deviation from the mean transpires through an impeding factor in the expression for $C_\pi(\gamma,\delta)$, quantifying the impact of asynchronicity in both mean and fluctuation terms.

TABLE III: Parameters of UCD-MXL0$^+$ of Corollary 2

a) $\gamma = \left[1 + \sqrt{\frac{2LK}{\lambda}d}\right]^{-1}\sqrt{\frac{\log M}{\lambda LKd}}$; $\delta = \frac{1}{2}\sqrt{\frac{LKd\log M}{\lambda}}$; $T \ge \frac{1}{4\lambda}[LKM^4\log M]$

b) $\gamma = \hat{\psi}(\alpha)\Big[\sqrt{\hat{\chi}(\alpha)L\lambda} + 2\log^{3/8}(\frac{1}{\alpha})\,\hat{\chi}(\alpha)\hat{\psi}(\alpha)\,L[Kd]^{3/4}\Big]^{-1}\Big(\frac{\log^{1/8}(1/\alpha)\sqrt{\log M}}{[Kd]^{3/4}}\Big)$;
$\delta = \Big(\frac{\hat{\psi}(\alpha)}{2}\sqrt{\frac{L}{\hat{\chi}(\alpha)\lambda}}\Big)[\sqrt{\log(1/\alpha)}\,Kd]^{1/4}\sqrt{\log M}$;
$T = 16L^2\Big[\hat{\psi}^2(\alpha) + \frac{\hat{\psi}(\alpha)\sqrt{\lambda/[\hat{\chi}(\alpha)L]}}{\log^{3/8}(1/\alpha)\,K^{3/4}d}\Big]^2\Big(\frac{\log(1/\alpha)K^6d^3\log M}{\varepsilon^2}\Big)$;
with $\hat{\psi}(\alpha) = \Big[\frac{\hat{\chi}(\alpha)}{\sqrt{K\sqrt{\log(1/\alpha)}}} + \sqrt{2/\log(M)}\,(1+1/K)\Big]^{1/2}$,
$\hat{\chi}(\alpha) = \Big[\sqrt{2}(1-1/K) + \frac{1}{2K\sqrt{\log(1/\alpha)}}\Big]^{1/2}$

In Appendix E, we show how the parameters (γ,δ) can be optimized for general Π; for concreteness, we present below the particular case where at any stage each user is active with probability $\pi_k = 1/K$:

Corollary 2 (Uniform AMXL0$^+$, $K \ge 2$)**.** *Suppose that AMXL0$^+$ is run with $\pi_1 = \cdots = \pi_K = 1/K$, $\gamma_t = \gamma/\sqrt{T}$, $\delta_t = \delta/\sqrt{T}$, and T, γ, δ as in Line 2. Then:*

a) In expectation, we have:

$$R^* - \mathbb{E}\left[R(\bar{\mathbf{Q}}_T)\right] \le 2L\left(1 + \sqrt{\lambda/L}\right)\sqrt{\frac{K^5M^6\log M}{T}}. \tag{35}$$

b) In probability, given a small enough tolerance $\varepsilon > 0$ and a confidence level $1-\alpha \in (0,1)$, we have:

$$\mathbb{P}(R^* - R(\bar{\mathbf{Q}}_{\lceil T\rceil}) \le \varepsilon) \ge 1-\alpha. \tag{36}$$

Remark 2 (Coordinate descent). The case $\Pi_{\{1\}} = \cdots = \Pi_{\{K\}} = 1/K$ where a single user is active at each time step with probability $\pi_k = 1/K$ covers the alternated optimization scheme known as (uniform) "coordinate descent" (UCD-MXL0$^+$)—the coordinates in this context refer to the wireless users. In this regard, Corollary 2 provides us with a quantification of the impact of alternation on the convergence speed of MXL0$^+$. Looking for instance at Corollaries 1(a) and 2(a), we observe that the expected convergence of the time average, if regarded as a function of the total number n of user updates, is $\mathcal{O}(\sqrt{K^5M^6\log M/n})$ both for the synchronized algorithm MXL0$^+$ and for UCD-MXL0$^+$. The impact of the network size K on the number of user updates needed for ε-convergence with probability $1-\alpha$, however, is more pronounced by an order of magnitude for UCD-MXL0$^+$, $\Theta\left(\log(1/\alpha)K^6M^6\log M/\varepsilon^2\right)$, than it is for MXL0$^+$, $\Theta\left(\log(1/\alpha)K^5M^6\log M/\varepsilon^2\right)$.

VI. Numerical Experiments

In this section, we perform a series of experiments to validate our results in realistic network conditions. Throughout what follows, and unless specified otherwise, our numerical experiments are performed in a simulated wireless network setup with parameters as summarized in Table IV. In more detail, we consider a cellular wireless network occupying a central frequency of $f_c = 2.5\,\text{GHz}$ and a total bandwidth of 10 MHz. Signal propagation in the wireless medium is modeled following the widely utilized COST 2100 channel model for moderately dense urban environments [34]. This is a geometry-based stochastic extension of the original COST Hata model

Parameter	Value
Time frame duration	5 ms
MIMO channel model	COST 2100 [34]
BS/MS antenna height	32 m / 1.5 m
Central frequency	2.5 GHz
Total bandwidth	11.2 MHz
Spectral noise density (20 °C)	−174 dBm/Hz
Maximum transmit power	$P = 33$ dBm
Transmit antennas per device	$M \in \{2, 4, 8\}$
Receive antennas	$N = 128$

TABLE IV: Wireless network simulation parameters.

[35] which has been designed to reproduce the stochastic properties of MIMO channels over the frequency, space and time domains. As such, even though it is not 5G-specific, the COST 2100 model is generic and flexible, making it suitable to model a broad range of multi-user or distributed MIMO scenarios [34].

Network coverage is provided by a base station (BS) with an effective service radius of 1 km (for the wider network in play, we consider a hexagonal cell coverage structure). The BS serves the uplink of K wireless transmitters that are positioned uniformly at random within the coverage area following a homogeneous Poisson point process. All communications occur over a TDD transmission scheme with an asynchronous frame duration of $T_f = 5$ ms. Finally, in line with state-of-the-art mobile and portable device specifications, transmitting devices are assumed to have a maximum transmit power of 33 dBm.

A. Comparison with water-filling methods

We begin by examining the performance of MXL-type methods relative to conventional water-filling schemes. To provide a broad basis for this comparison, we focus on two complementing scenarios: (*a*) the *full feedback* case, i.e., when transmitters are assumed to know their individual channel matrices $\mathbf{H}_k$ and the induced signal-plus-noise covariance matrix $\mathbf{W}$; and (*b*) the *limited feedback* case, i.e., when transmitters only observe their realized utility (i.e., their sum rate). For the purposes of our experiments, and in line with other recent works on large antenna arrays [1, 5, 6, 36], we consider a system with $K = 60$ users, each with 2, 4 or 8 transmit antennas, and a BS with $N = 128$ receive antennas; all other network parameters are as in Table IV.

In the first case (full matrix feedback), we simulated the iterative and simultaneous variants of water-filling against the MXL algorithm as presented in Section II-C. The iterative WF variant converges to an optimum solution; however, because user updates need to be taken in a sequential, round-robin fashion, the algorithm's convergence speed is inversely proportional to the number of users in the system, and hence quite slow. On the other hand, the simultaneous WF variant achieves significant performance gains within the first few iterations, but because it has no way of mitigating conflicting user updates, these gains subsequently evaporate and the algorithm converges to a suboptimal state. By comparison, the MXL algorithm achieves convergence to an optimal state within a few iterations, without suffering from the slow convergence speed of the iterative WF algorithm or the oscillatory behavior of its simultaneous counterpart. The results of these simulations are plotted in Fig. 1a.

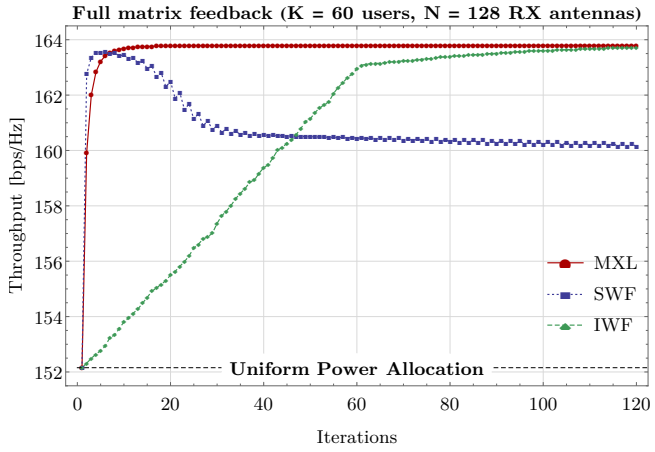
Moving forward, to establish a fair comparison in the limited feedback case, we consider a baseline setting where, at each transmission frame $t = 1, 2 \dots$, each user has access to one-point pilot estimates of their effective channel matrix $\tilde{\mathbf{H}}_k = \mathbf{H}_k \mathbf{W}_k^{-1/2}$ (e.g., via randomized directional sampling) [19]. Since $\mathbf{W}_k \equiv \mathbf{W}_k(\mathbf{Q}_t)$ evolves over time (because of the signal covariance modulation $\mathbf{Q}_t$ of all other users in the network), these measurements must be repeated over time; otherwise, knowledge of $\mathbf{H}_k$ alone would not suffice to run water-filling in a multi-user environment. By comparison, for the MXL0$^+$ algorithm, we only assume that users observe their realized throughput as described in detail in Section IV.

The results of our simulations are plotted in Fig. 1b. Because water-filling methods require perfect knowledge of $\tilde{\mathbf{H}}_k$ at each transmission frame, the imperfections introduced by one-point pilot contamination effects cause a complete breakdown of the algorithm's convergence. In particular, both iterative and simultaneous variants fail to exhibit any significant performance gains over a uniform (isotropic) input signal covariance profile. The performance of MXL0 is underwhelming in the first iterations (due to exploration), but it improves steadily over time; however, this improvement is very slow over the simulation window. On the other hand, the callback mechanism of MXL0$^+$ achieves dramatically better results, even with one-shot, zeroth-order feedback.

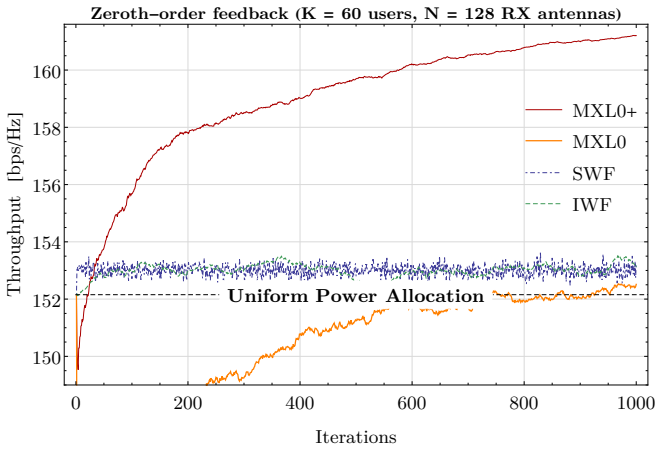
In terms of per-iteration computational complexity, Fig. 2 compares the wall-clock runtime of an iteration of each algorithm (IWF, SWF and MXL0$^+$). All computations were performed in a commercial laptop with 16 GB RAM and a 2.6 GHz 6-core Intel i7 CPU; for statistical significance, they were averaged over $S = 1000$ sample runs. Network parameters were as above, except for the number of receive antennas which was taken in the range $\{4, \dots, 64\}$ to assess scalability. For small values of N, IWF has the fastest runtime per iteration because only one user updates per iteration and the inversion of the MUI matrix at the receiver is relatively fast. However, for larger values of N, this advantage evaporates and MXL0$^+$ becomes the fastest because the SPSA estimator is sparse, so the resulting matrix operations are the lightest. This provides an additional layer to the results of Fig. 1: even though IWF/SWF methods fail to produce any measurable performance gains in limited feedback environments, MXL0$^+$ remains optimal and achieves considerably better throughput values, all with a lighter per-iteration runtime.

B. Convergence speed analysis

For completeness, we also examine below the convergence speed of the different MXL methods with limited, zeroth-order feedback. The results of our experiments are reported in Fig. 3 where we plot the users' relative distance to optimality in a log-log scale under the three gradient-free algorithms discussed in the previous sections, MXL0, MXL0$^+$ and AMXL0$^+$ (Algs. 1—3 respectively, the third in the coordinate descent form UCD-MXL0$^+$ discussed in Remark 2). We plot the relative ratio $\rho = (R^* - R(\bar{\mathbf{Q}}_t))/(R^* - R_1)$, so $\rho = 1$ corresponds to the initialization of each algorithm while $\rho = 0$ corresponds to optimality. All algorithms were run with constant step size and



(a) Full matrix feedback



(b) Limited, zeroth-order feedback

Fig. 1: Comparison between water-filling and matrix exponential learning in a wireless network with $K = 60$ users and $N = 128$ receive antennas. In the full feedback case (left), the transmitters are running MXL against IWF/SWF with full matrix information (perfect knowledge of effective channel matrices, system-wide signal-plus-noise covariance matrix, etc.). In the zeroth-order case (right), the transmitters only have access to their realized utility (the achieved throughput) and are running MXL0 and MXL0$^+$ against IWF/SWF with one-shot pilot estimates of the required matrix information. In both instances, MXL/MXL0$^+$ exhibits consistent – and significant – performance gains over WF methods.

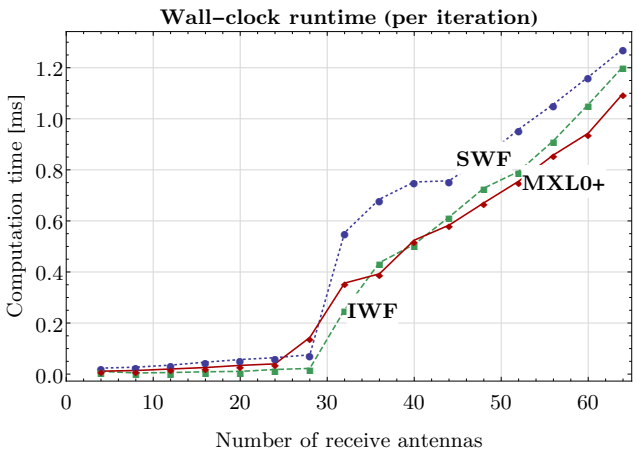


Fig. 2: Per-iteration runtime of the IWF, SWF and MXL0$^+$ algorithms as a function of the number of antennas at the receiver (lower is better).

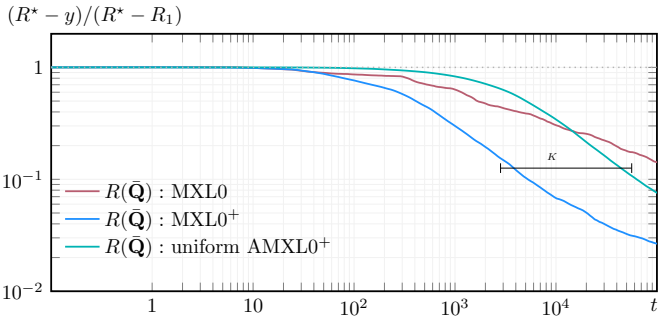


Fig. 3: Convergence speed of the proposed methods ($N = 16$, $K = 20$). The callback in MXL0$^+$ greatly improves performance over MXL0.

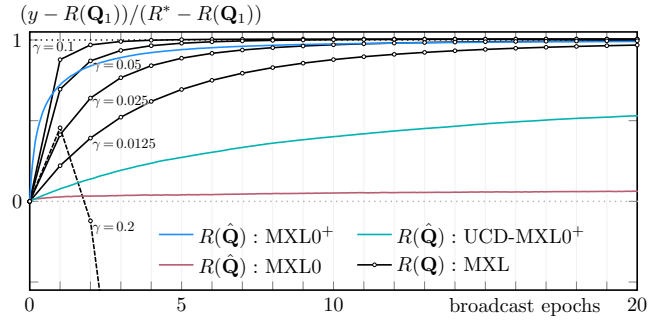


Fig. 4: Overhead of the algorithms under study in a large network ($K = 50$, $N = 128$). When normalized for overhead, MXL0$^+$ matches the performance of finely tuned gradient-based methods.

query radius in a system with $K = 20$ users. Despite the severe feedback limitations, we see that (UCD-)MXL0$^+$ rapidly closes the initial optimality gap (in line with Fig. 1b).

A close inspection of the slopes of the various curves on the log-log graph further reveals the $\mathcal{O}(1/\sqrt[4]{t})$ complexity of MXL0 and the $\mathcal{O}(1/\sqrt{t})$ complexity of (UCD-)MXL0$^+$, in full accordance with Theorems 2, 4 and 5. The log K shift between UCD-MXL0$^+$ and MXL0$^+$ predicted in Remark 2 can also be clearly observed.

Finally, Fig. 4 provides a normalized comparison to gradient-based methods in a network with $N = 128$ receive antennas and $K = 50$ users. Here, access to full matrix feedback would require $N^2 = 32$ MB of 16-bit data per frame; in view of this, we examine instead the algorithms' convergence speed in terms of the feedback epochs required for convergence. For benchmarking purposes, we ran MXL with a constant step-size (the most principled choice given the smoothness of R). Quite remarkably, we see that MXL0$^+$ remains competitive with—and even outperforms!—the fastest implementations of MXL. On the other hand, UCD-MXL0$^+$ was approximately $K = 50$ times slower than MXL0$^+$, while MXL0 was essentially non-convergent.

VII. Discussion

In this paper, we proposed a series of online optimization schemes for *distributed, feedback-limited* multi-user MIMO systems that circumvent the need for matrix feedback (perfect, noisy, or otherwise). Gradient estimation methods based on

conventional simultaneous perturbation stochastic approximation (SPSA) techniques lead to an $\mathcal{O}(1/T^{1/4})$ convergence rate, which is catastrophically slow for large MIMO systems. To overcome this deficiency, we introduced an acceleration mechanism which achieves an $\mathcal{O}(1/T^{1/2})$ convergence rate through the reuse of previous throughput measurements. In this regard, the proposed MXL0$^+$ algorithm enjoys the best of many worlds: it achieves convergence with minimal feedback requirements (a single scalar), it matches the convergence speed of conventional methods that require full mtrix feedback, all the while remaining simple in principle and easy to implement.

Although we focused on the throughput maximization problem in the single-cell MIMO multiple-access channel, our proposed algorithms can also be applied to multi-cell networks operating in orthogonal frequency bands so that the inter-cell interference is canceled; the sum rate in each cell can be optimized separately and independently without any loss of global optimality. In dense small-cell networks, in which the interference cannot be canceled this way, the network sum-rate optimization problem is a known difficult non-convex problem [15, 16]. A possible workaround is to consider autonomous small-cells that aim at maximizing their own sum rate (similar in spirit to (Opt$_k$) in Section II-B), which leads to a concave non-cooperative game. In our previous work [37], we showed that the original MXL converges to the Nash equilibrium solution of such games under milder assumptions compared to iterative water-filling; studying the performance of our gradient-free algorithms MXL0 and MXL0$^+$ in such settings is an interesting and non-trivial extension of the present work.

Moving beyond throughput maximization, the gradient-free methodology presented in this work can also be tailored to a wide range of resource allocation problems that arise in signal processing and wireless communications (from power control to energy efficiency). For example, by using the Charnes-Cooper transformation to turn non-convex fractional optimization problems into convex ones [38], the material developed in this paper can be applied to the core problem of energy-efficiency maximization problem in multi-user MIMO systems. These applications, which are deferred to future work, highlight the potential of the gradient-free algorithms derived here.

Finally, in terms of practical implementation, we should note that our analysis provides precise computational complexity and runtime bounds; however, it does not address the processing power expenditure on "off-the-shelf" wireless devices. Investigating this aspect of the proposed methods is a very fruitful research direction which we intend to address in future work.

Appendix
Technical Proofs

A. Matrix exponential learning as a dual averaging scheme

In our developments, the space of the covariance matrices of each user is equipped with the nuclear norm, given for any Hermitian matrix $\mathbf{Q}$ by $\|\mathbf{Q}\| = \mathrm{tr}(\sqrt{\mathbf{Q}\mathbf{Q}})$, and equivalent to the L_1-norm of the vector of the eigenvalues of $\mathbf{Q}$. The dual of the nuclear norm, $\|\mathbf{Q}\|_* = \max_{\mathbf{Q}'}\{\mathrm{tr}(\mathbf{Q}\mathbf{Q}') : \|\mathbf{Q}'\| \leq 1\}$, reduces to the L_∞-norm of the vector of eigenvalues. For every $m \times m$ Hermitian matrix $\mathbf{Q}$, one has

$$\|\mathbf{Q}\|_* \leq \|\mathbf{Q}\|_2 \leq \|\mathbf{Q}\| \leq \sqrt{m}\|\mathbf{Q}\|_2 \leq m\|\mathbf{Q}\|_*, \tag{A.1}$$

where $\|\mathbf{Q}\|_2 = \sqrt{\mathrm{tr}(\mathbf{Q}\mathbf{Q})}$ denotes the (Frobenius) L^2-norm of $\mathbf{Q}$. From the global perspective of matrix arrangements $\mathbf{Q} = (\mathbf{Q}_1, ..., \mathbf{Q}_K)$—now regarded as block diagonal covariance matrices—, the trace norm and its dual naturally extend as

$$\|\mathbf{Q}\| = \textstyle\sum_{k=1}^K \|\mathbf{Q}_k\|, \qquad \|\mathbf{Q}\|_* = \max_{k\in\{1,...,K\}} \|\mathbf{Q}_k\|_*. \tag{A.2}$$

We now derive the matrix exponential learning step and some properties of it. To this end, we place ourselves in the compact set $\mathcal{Q} = \{\mathbf{Q} \in \mathrm{Herm}(M) : \mathrm{tr}(\mathbf{Q}) = 1, \mathbf{Q} \succeq 0\}$ of the M-dimensional positive semidefinite Hermitian matrices with unit trace—the parameter M stands for the number of antennas of any of the K users. Let the inner product $\langle \mathbf{Y}, \mathbf{Q}\rangle = \mathrm{tr}(\mathbf{Y}\mathbf{Q})$ denote the value at $\mathbf{Q} \in \mathcal{Q}$ of the linear function induced by $\mathbf{Y} \in \mathcal{Y}$, where $\mathcal{Y} = \{\mathbf{Z} \in \mathrm{Herm}(M) : \mathrm{tr}(\mathbf{Z}) = 0\}$ is tangent to $\mathcal{Q}$. For any differentiable function f on $\mathrm{Herm}(M)$, we denote by $\nabla f : \mathcal{Q} \mapsto \mathcal{Y}$ the *orthogonal projection* of the gradient ∇f on the tangent space $\mathcal{Y}$, given by $\nabla f = \nabla f - \mathrm{tr}(\nabla f)\,\mathbf{I}$.

Lemma A.1.

(i) The regularization function[5] $h(\mathbf{Q}) = \mathrm{tr}(\mathbf{Q}\log\mathbf{Q})$ *is* 1*-strongly convex over* $\mathcal{Q}$ *with respect to* $\|\cdot\|$.

(ii) The conjugate of h, $h^* : \mathcal{Y} \mapsto \mathbb{R}$*, defined by*

$$h^*(\mathbf{Y}) = \max_{\mathbf{Q}\in\mathcal{Q}}\{\langle \mathbf{Y}, \mathbf{Q}\rangle - h(\mathbf{Q})\}, \tag{A.3}$$

is differentiable with gradient $\nabla h^* = \mathbf{\Lambda}$*, where* $\mathbf{\Lambda}$ *is the exponential learning mapping defined by*

$$\mathbf{\Lambda}(\mathbf{Y}) = \frac{\exp(\mathbf{Y})}{\mathrm{tr}(\exp(\mathbf{Y}))}. \tag{A.4}$$

(iii) For $\mathbf{Q} \in \mathcal{Q}$ *and* $\mathbf{Y} \in \mathcal{Y}$*,*

$$\mathbf{Q} = \mathbf{\Lambda}(\mathbf{Y}) \Leftrightarrow \mathbf{Y} = \nabla h(\mathbf{Q}). \tag{A.5}$$

(iv) h^* *is* 1*-smooth with respect to the dual norm* $\|\cdot\|_*$.

Proof. We refer to [39] for the strong convexity of h. For (ii), the differentiablity of h^* is a consequence of Danskin's theorem (e.g. [40]), which, besides, gives us the gradient of (A.3),

$$\nabla h^*(\mathbf{Y}) = \arg\max_{\mathbf{Q}\in\mathcal{Q}}\{\langle \mathbf{Y}, \mathbf{Q}\rangle - h(\mathbf{Q})\}. \tag{A.6}$$

Relaxing the constraint $\mathrm{tr}(\mathbf{Q}) - 1 = 0$ in the subproblem (A.6) and using $\nabla h(\mathbf{Q}) = \mathbf{I} + \log\mathbf{Q}$ yields the stationarity condition

$$\log\mathbf{Q} - \mathbf{Y} + (1+\nu)\,\mathbf{I} = 0, \tag{S}$$

where $\nu \in \mathbb{R}$ is the Lagrange multiplier related to the constraint. Condition (S) rewrites as $\mathbf{Q} = \exp(-(1+\nu))\exp(\mathbf{Y})$, which implies the primal feasibility condition $\mathbf{Q} \succeq 0$. The remaining KKT conditions $\mathrm{tr}(\mathbf{Q}) - 1 \leq 0$ and $\nu(\mathrm{tr}(\mathbf{Q}) - 1) = 0$ yield $\nu = \log(\mathrm{tr}(\exp(\mathbf{Y}))) - 1$, and $\mathbf{Q} = \mathbf{\Lambda}(\mathbf{Y})$ as the unique maximizer of (A.6), which completes the proof of (ii).

Now, it follows from (A.6) that, for any $\mathbf{Y} \in \mathcal{Y}_k$, one has $\mathbf{Q} = \mathbf{\Lambda}(\mathbf{Y})$ if and only if $\langle \mathbf{Y}, \mathbf{Q}'\rangle - h(\mathbf{Q}') \leq \langle \mathbf{Y}, \mathbf{Q}\rangle - h(\mathbf{Q})$ holds for all $\mathbf{Q}' \in \mathcal{Q}$, i.e., iff $\mathbf{Y}$ is a subgradient of h at $\mathbf{Q}$. Claim (iii) follows by differentiability of h.

[5] We use here the convention $0\log 0 = 0$.

Finally, (iv) is a property of convex conjugation [41]. Indeed, let $\mathbf{Y}, \mathbf{Y}' \in \mathcal{Y}$ and $\mathbf{Q} = \mathbf{\Lambda}(\mathbf{Y})$. By convexity,

$$h(\mathbf{Q}') \geq h(\mathbf{Q}) + \langle \nabla h(\mathbf{Q}), \mathbf{Q}' - \mathbf{Q} \rangle + \tfrac{1}{2}\|\mathbf{Q}' - \mathbf{Q}\|^2 \tag{A.7}$$

holds for any $\mathbf{Q}' \in \mathcal{Q}$. It follows that

$$\begin{aligned} h^*(\mathbf{Y}') &\overset{(A.3)}{=} \max_{\mathbf{Q}' \in \mathcal{Q}} \{\langle \mathbf{Y}', \mathbf{Q}' \rangle - h(\mathbf{Q}')\} \\ &\overset{(A.7)}{\leq} \max_{\mathbf{Q}' \in \mathcal{Q}} \{\langle \mathbf{Y}', \mathbf{Q}' \rangle - h(\mathbf{Q}) - \langle \nabla h(\mathbf{Q}), \mathbf{Q}' - \mathbf{Q} \rangle - \tfrac{1}{2}\|\mathbf{Q}' - \mathbf{Q}\|^2\} \\ &\overset{(A.5)}{=} \langle \mathbf{Y}, \mathbf{Q} \rangle - h(\mathbf{Q}) + \langle \mathbf{Y}' - \mathbf{Y}, \mathbf{Q} \rangle + \max_{\mathbf{Q}' \in \mathcal{Q}} \{\langle \mathbf{Y}' - \mathbf{Y}, \mathbf{Q}' - \mathbf{Q} \rangle \\ &\qquad - \tfrac{1}{2}\|\mathbf{Q}' - \mathbf{Q}\|^2\} \\ &\overset{(A.6)}{\leq} h^*(\mathbf{Y}) + \langle \mathbf{Y}' - \mathbf{Y}, \nabla h^*(\mathbf{Y}) \rangle + \tfrac{1}{2}\|\mathbf{Y}' - \mathbf{Y}\|_*^2 \end{aligned} \tag{A.8}$$

and h^* is 1-smooth. Equivently, (A.8) rewrites as[42]

$$\|\mathbf{\Lambda}(\mathbf{Y}) - \mathbf{\Lambda}(\mathbf{Y}')\| \leq \|\mathbf{Y} - \mathbf{Y}'\|_* \quad \forall \mathbf{Y}, \mathbf{Y}' \in \mathcal{Y}_k. \tag{A.9}$$

■

We now consider the Fenchel primal-dual coupling $F : \mathcal{Q} \times \mathcal{Y} \mapsto \mathbb{R}$ associated with the entropic regularizer h.

Lemma A.2. *The Fenchel coupling*

$$F(\mathbf{Q}, \mathbf{Y}) = h(\mathbf{Q}) + h^*(\mathbf{Y}) - \langle \mathbf{Y}, \mathbf{Q} \rangle \tag{A.10}$$

satisfies the following properties. For $\mathbf{Q} \in \mathcal{Q}$ and $\mathbf{Y}, \mathbf{Y}' \in \mathcal{Y}$,

$$F(\mathbf{Q}, \mathbf{Y}') \leq F(\mathbf{Q}, \mathbf{Y}) + \langle \mathbf{Y}' - \mathbf{Y}, \mathbf{\Lambda}(\mathbf{Y}) - \mathbf{Q} \rangle + \tfrac{1}{2}\|\mathbf{Y}' - \mathbf{Y}\|_*^2, \tag{A.11a}$$

$$F(\mathbf{Q}, \mathbf{Y}) \geq \tfrac{1}{2}\|\mathbf{Q} - \mathbf{\Lambda}(\mathbf{Y})\|^2, \tag{A.11b}$$

$$F(\mathbf{Q}, \mathbf{Y}) \geq 0 \text{ with } F(\mathbf{Q}, \mathbf{Y}) = 0 \Leftrightarrow \mathbf{Y} = \nabla h(\mathbf{Q}). \tag{A.11c}$$

Proof. Equations (A.11a) and (A.11b) follow from the smoothness of h^* and from the strong convexity of h, respectively. Indeed, we get (A.11a) by combining (A.10) with (A.8), while

$$\begin{aligned} F(\mathbf{Q}, \mathbf{Y}) &\overset{(A.10)}{=} \max_{\mathbf{Q}' \in \mathcal{Q}} \{h(\mathbf{Q}) - h(\mathbf{Q}') - \langle \mathbf{Y}, \mathbf{Q} - \mathbf{Q}' \rangle\} \\ &\overset{(A.5)}{\geq} h(\mathbf{Q}) - h(\mathbf{\Lambda}(\mathbf{Y})) - \langle \nabla h(\mathbf{\Lambda}(\mathbf{Y})), \mathbf{Q} - \mathbf{\Lambda}(\mathbf{Y}) \rangle \\ &\overset{(A.7)}{\geq} \tfrac{1}{2}\|\mathbf{Q} - \mathbf{\Lambda}(\mathbf{Y})\|^2 \end{aligned} \tag{A.12}$$

yields (A.11b). Then, (A.11c) follows from (A.11b) and (A.5). ■

B. The SPSA estimator

This section is concerned with the bias of the gradient estimator defined, for $k = 1, \ldots, K$, by

$$\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho) = \tfrac{d_k}{\delta}[R(\hat{\mathbf{Q}}) - \rho]\, \mathbf{Z}_k, \tag{B.1}$$

where $\delta > 0$ is a given query radius, $\hat{\mathbf{Q}} = (\hat{\mathbf{Q}}_1, \ldots, \hat{\mathbf{Q}}_K)$ is given by (20), $\mathbf{Z} = (\mathbf{Z}_1, \ldots, \mathbf{Z}_K)$ with $\mathbf{Z}_k$ sampled uniformly on the sphere $\mathbb{S}^{d_k - 1}$, and ρ an arbitrary scalar offset quantity independent of $\mathbf{Z}$. Observe that (B.1) covers the gradient estimators of both MXL0 and MXL0$^+$.

The computation of a bound for the bias of estimator (B.1) is based on Stokes' theorem, applied to the sphere $\mathbb{S}^{d_k - 1}$:

$$\int_{\mathbb{S}^{d_k - 1}} f(\mathbf{Z}_k)\, \mathbf{Z}_k\, d\mu(\mathbf{Z}_k) = \int_{\mathbb{B}^{d_k}} \nabla f(\boldsymbol{\zeta})\, d\mu(\boldsymbol{\zeta}), \tag{B.2}$$

where f is any function on $\mathrm{Herm}(M_k)$ and μ denotes the Lebesgue measure. Before proceeding, observe that each test covariance marix $\hat{\mathbf{Q}}_k$ is bound to the initial matrix $\mathbf{Q}_k$ by $\|\hat{\mathbf{Q}}_k - \mathbf{Q}_k\|_2 \leq 2\delta \|\mathbf{Z}_k\|_2$, where, under our assumption $d_k > 0$, $\|\mathbf{Z}_k\|_* \leq 1/2$ for every $\mathbf{Z}_k \in \mathbb{S}^{d_k - 1}$. It follows from (A.1) that any test configuration $\hat{\mathbf{Q}}$ in (SPSA) and (SPSA+) satisfies

$$\|\hat{\mathbf{Q}} - \mathbf{Q}\|_2 \leq 2\delta K, \quad \text{and} \quad \|\hat{\mathbf{Q}} - \mathbf{Q}\| \leq 2\delta K \sqrt{d}. \tag{B.3}$$

Lemma B.1. *The estimator* (B.1) *satisfies*

$$\|\mathbb{E}[\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho) - \nabla_k R(\mathbf{Q})]\|_* \leq 2K\lambda_k\, \delta, \tag{B.4}$$

$$\|\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho)\|_* \leq \tfrac{d_k}{2^K \delta} \max_{\mathbf{Q}' \in \mathcal{Q}} |R(\mathbf{Q}') - \rho|. \tag{B.5}$$

Proof of Lemma B.1. We argue as in [21, 43]. By introducing the notation $\tilde{\mathbf{Q}}^\delta(\boldsymbol{\zeta}) = (\hat{\mathbf{Q}}_1, \ldots, \hat{\mathbf{Q}}_{k-1}, \tilde{\mathbf{Q}}_k^\delta(\boldsymbol{\zeta}), \hat{\mathbf{Q}}_{k+1}, \ldots, \hat{\mathbf{Q}}_K)$, in which $\tilde{\mathbf{Q}}_k^\delta(\boldsymbol{\zeta}) = \mathbf{Q}_k + (\delta/r_k)(\mathbf{C}_k - \mathbf{Q}_k) + \delta\boldsymbol{\zeta}$, we find

$$\begin{aligned} &\|\mathbb{E}[\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho) - \nabla_k R(\mathbf{Q})]\|_* \\ &\quad \overset{(B.1)}{=} \left\| \mathbb{E}[\tfrac{d_k}{\delta}[R(\hat{\mathbf{Q}}) - \rho]\, \mathbf{Z}_k - \nabla_k R(\mathbf{Q})] \right\|_* \\ &\quad = \left\| \tfrac{d_k}{\delta} \mathbb{E}[R(\hat{\mathbf{Q}})\, \mathbf{Z}_k] - \nabla_k R(\mathbf{Q}) \right\|_* \\ &\quad = \left\| \tfrac{d_k}{\delta} \mathbb{E}\left[\frac{\int_{\mathbb{S}^{d_k - 1}} R(\tilde{\mathbf{Q}}^\delta(\mathbf{Z}_k))\, \mathbf{Z}_k\, d\mu(\mathbf{Z}_k)}{\mathrm{vol}(\mathbb{S}^{d_k - 1})} \right] - \nabla_k R(\mathbf{Q}) \right\|_* \\ &\quad \overset{(20)}{=} \left\| \mathbb{E}\left[\frac{\int_{\mathbb{S}^{d_k - 1}} R(\tilde{\mathbf{Q}}^\delta(\mathbf{Z}_k))\, \mathbf{Z}_k\, d\mu(\mathbf{Z}_k)}{\delta \mathrm{vol}(\mathbb{B}^{d_k})} - \nabla_k R(\mathbf{Q}) \right] \right\|_*. \end{aligned} \tag{B.6}$$

It follows from Stokes' theorem that (B.6) reduces to

$$\begin{aligned} &\|\mathbb{E}[\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho) - \nabla_k R(\mathbf{Q})]\|_* \\ &\quad \overset{(B.2)}{=} \left\| \mathbb{E}\left[\frac{\int_{\mathbb{B}^{d_k}} \delta \nabla_k R(\tilde{\mathbf{Q}}^\delta(\boldsymbol{\zeta}))\, d\mu(\boldsymbol{\zeta})}{\delta \mathrm{vol}(\mathbb{B}^{d_k})} - \nabla_k R(\mathbf{Q}) \right] \right\|_* \\ &\quad \leq \mathbb{E}\left[\tfrac{1}{\mathrm{vol}(\mathbb{B}^{d_k})} \int_{\mathbb{B}^{d_k}} \|\nabla_k R(\tilde{\mathbf{Q}}^\delta(\boldsymbol{\zeta})) - \nabla_k R(\mathbf{Q})\|_*\, d\mu(\boldsymbol{\zeta}) \right] \\ &\quad \overset{(13)}{\leq} \frac{\int_{\mathbb{B}^{d_k}} \lambda_{kk} \|\tilde{\mathbf{Q}}_k^\delta(\boldsymbol{\zeta}) - \mathbf{Q}_k\|_2\, d\mu(\boldsymbol{\zeta})}{\mathrm{vol}(\mathbb{B}^{d_k})} + \textstyle\sum_{\ell \neq k} \lambda_{k\ell} \mathbb{E}[\|\hat{\mathbf{Q}}_\ell - \mathbf{Q}_\ell\|_2] \\ &\quad \overset{(20)}{\leq} \lambda_{kk}\left(1 + \frac{\int_{\mathbb{B}^{d_k}} \|\boldsymbol{\zeta}\|_2\, d\mu(\boldsymbol{\zeta})}{\mathrm{vol}(\mathbb{B}^{d_k})}\right)\delta + 2\textstyle\sum_{\ell \neq k} \lambda_{k\ell} \mathbb{E}\left[\|\mathbf{Z}_\ell\|_2\right]\delta \\ &\quad \leq \lambda_{kk}(\tfrac{2M_k + 1}{M_k + 1})\delta + 2\textstyle\sum_{\ell \neq k} \lambda_{k\ell}\delta \;\leq\; 2K\lambda_k\delta, \end{aligned} \tag{B.7}$$

and we recover (B.4). Then (B.5) is immediate from the definition of $\mathbf{V}_k(\mathbf{Q}, \mathbf{Z}; \rho)$ and the fact that $\|\mathbf{Z}_\ell\|_* \leq 1/2$ for all ℓ. ■

C. Analysis of the MXL0 algorithm

Let $\mathcal{Q}^*$ denote the solution set of (Opt). Given any $\mathbf{Q}^* \in \mathcal{Q}^*$, we consider, for analysis purposes, the Lyapunov function

$$\mathcal{L}(\mathbf{Y}; \mathbf{Q}^*) = \textstyle\sum_{k=1}^K F(\mathbf{Q}_k^*, \mathbf{Y}_k), \tag{C.1}$$

where F is the Fenchel coupling defined in (A.10). If $\mathcal{F}_{t-1} = (\mathbf{Y}_1, \mathbf{Z}_1, \ldots, \mathbf{Y}_{t-1}, \mathbf{Z}_{t-1})$ denotes the history of MXL0 up to step $t - 1$, the gradient estimator (B.1) decomposes into

$$\mathbf{V}_{k,t} = \nabla_k R(\mathbf{Q}_t) + \mathbf{B}_{k,t} + \mathbf{U}_{k,t}, \tag{C.2}$$

where $\mathbf{B}_{k,t} = \mathbb{E}[\mathbf{V}_{k,t} | \mathcal{F}_{t-1}] - \nabla_k R(\mathbf{Q}_t)$ is the systematic error on $\mathbf{V}_{k,t}$, bounded by

$$\|\mathbf{B}_{k,t}\|_* \overset{(B.4)}{\leq} 2K\lambda_k\delta_t, \tag{C.3}$$

and $\mathbf{U}_{k,t}$ is the random deviation of $\mathbf{V}_{k,t}$ from its expected value $\mathbb{E}[\mathbf{V}_{k,t} | \mathcal{F}_{t-1}]$, so that $\mathbb{E}[\mathbf{U}_{k,t} | \mathcal{F}_{t-1}] = 0$, and

$$\|\mathbf{U}_{k,t}\|_* \leq \|\mathbf{V}_{k,t}\|_* + \mathbb{E}[\|\mathbf{V}_{k,t}\|_* | \mathcal{F}_{t-1}]. \tag{C.4}$$

In our analysis we consider the following random sequence:

$$Z_t = \gamma_t \textstyle\sum_{k=1}^K \langle \mathbf{U}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^* \rangle. \tag{C.5}$$

Since $|\langle \mathbf{U}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^* \rangle| \leq \|\mathbf{U}_{k,t}\|_* \|\mathbf{Q}_{k,t} - \mathbf{Q}_k^*\| \leq 2\|\mathbf{U}_{k,t}\|_*$, one has

$$\text{a)} \quad \mathbb{E}[Z_t | \mathcal{F}_{t-1}] = 0, \qquad \text{b)} \quad |Z_t| \leq 2\gamma_t \textstyle\sum_{k=1}^K \|\mathbf{U}_{k,t}\|_*. \tag{C.6}$$

Lemma C.1. *Run MXL0/MXL0$^+$ for t iterations under* (H0).

(i) With any step-size and query radius policy (γ_t, δ_t),

$$\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*) \le \mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) - \gamma_t\,[R^* - R(\mathbf{Q}_t)] + Z_t \\ +4K^2\lambda\gamma_t\delta_t + \tfrac{\gamma_t^2}{2}\textstyle\sum_{k=1}^K \|\mathbf{V}_{k,t}\|_*^2 \tag{C.7}$$

holds for $\mathbf{Q}^* \in \mathcal{Q}^*$, where the sequence Z_t is defined as in (C.5).

(ii) With decreasing policy $(\gamma_t, \delta_t) = (\tilde\gamma\, t^{-\alpha}, \tilde\delta\, t^{-\beta})$, such that $\alpha, \beta \ge 0$ and $\tilde\gamma, \tilde\delta > 0$,

$$R^* - \mathbb{E}[R(\bar{\mathbf{Q}}_t)] \le \tfrac{\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)}{\tilde\gamma\sum_{s=1}^t s^{-\alpha}} + \tfrac{4K^2\lambda\tilde\delta\sum_{s=1}^t s^{-\alpha-\beta}}{\sum_{s=1}^t s^{-\alpha}} \\ + \tfrac{\tilde\gamma\sum_{s=1}^t s^{-2\alpha}\sum_{k=1}^K \|\mathbf{V}_{k,s}\|_*^2}{2\sum_{s=1}^t s^{-\alpha}}. \tag{C.8}$$

(iii) With constant policy $(\gamma_t, \delta_t) = (\tilde\gamma, \tilde\delta)$, such that $\tilde\gamma, \tilde\delta > 0$,

$$R^* - \mathbb{E}[R(\bar{\mathbf{Q}}_T)] \le \tfrac{\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)}{T\tilde\gamma} + 4K^2\lambda\tilde\delta + \tfrac{\tilde\gamma\sum_{t=1}^T\sum_{k=1}^K \|\mathbf{V}_{k,t}\|_*^2}{2T} \tag{C.9}$$

for any $T \ge 1$. Further, if there exists $\bar v > 0$ such that $\|\mathbf{V}_{k,t}\|_* \le d\bar v$ for $k = 1,\dots,K$ and $t = 1,\dots,T$, then

$$\mathbb{P}\big(\tfrac{1}{T\tilde\gamma}\textstyle\sum_{t=1}^T Z_t \le \varepsilon\big) \ge 1 - \exp\big(-\tfrac{T\varepsilon^2}{32\bar v^2 d^2}\big). \tag{C.10}$$

Proof of Lemma C.1. (i) If $\mathbf{Q}^* \in \mathcal{Q}^*$, the concavity of R gives

$$\textstyle\sum_{k=1}^K \langle \nabla_k R(\mathbf{Q}), \mathbf{Q}_k - \mathbf{Q}_k^*\rangle \le R(\mathbf{Q}) - R^*, \quad \forall \mathbf{Q} \in \mathcal{Q}. \tag{C.11}$$

It follows from Lemma A.2 that

$$\begin{aligned}\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*) &\overset{(\text{MXL})}{=} \textstyle\sum_{k=1}^K F(\mathbf{Q}_k^*, \mathbf{Y}_{k,t} + \gamma_t\mathbf{V}_{k,t}) \\ &\overset{(\text{A.11a})}{\le} \mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) + \textstyle\sum_{k=1}^K \Big[\gamma_t\langle \mathbf{V}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle + \tfrac{\gamma_t^2}{2}\|\mathbf{V}_{k,t}\|_*^2\Big] \\ &\overset{(\text{C.2})}{=} \mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) + \gamma_t\textstyle\sum_{k=1}^K\langle \nabla_k R(\mathbf{Q}_t), \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle + Z_t \\ &\quad + \textstyle\sum_{k=1}^K \Big[\gamma_t\langle \mathbf{B}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle + \tfrac{\gamma_t^2}{2}\|\mathbf{V}_{k,t}\|_*^2\Big],\end{aligned} \tag{C.12}$$

Besides, (C.3) gives $|\langle \mathbf{B}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle| \le 4K\lambda_k\delta_t$, which combined with (C.11) and (C.12) yields Inequality (C.7).

(ii) By telescoping (C.7) $t-1$ times, dividing by $\sum_{s=1}^t \gamma_t$, and using $\mathcal{L}(\mathbf{Y}_{n+1};\mathbf{Q}^*) \ge 0$, we find

$$R^* - \tfrac{\sum_{s=1}^t \gamma_s R(\mathbf{Q}_s)}{\sum_{s=1}^t \gamma_s} \le \tfrac{\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)}{\sum_{s=1}^t \gamma_s} + \tfrac{\sum_{s=1}^t Z_s}{\sum_{s=1}^t \gamma_s} \\ +4K^2\lambda\tfrac{\sum_{s=1}^t \gamma_s\delta_s}{\sum_{s=1}^t \gamma_s} + \tfrac{1}{2}\tfrac{\sum_{s=1}^t \gamma_s^2\sum_{k=1}^K \|\mathbf{V}_{k,s}\|_*^2}{\sum_{s=1}^t \gamma_s}. \tag{C.13}$$

By concavity of R, the time average of the estimates satisfies

$$R(\bar{\mathbf{Q}}_t) \ge \big(\tfrac{1}{\sum_{s=1}^t \gamma_s}\big)\textstyle\sum_{s=1}^t \gamma_s R(\mathbf{Q}_s). \tag{C.14}$$

Introducing the suggested policies in (C.13) and using (C.14) gives

$$R^* - R(\bar{\mathbf{Q}}_t) \le \tfrac{\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)}{\tilde\gamma\sum_{s=1}^t s^{-\alpha}} + \tfrac{\sum_{s=1}^t Z_s}{\tilde\gamma\sum_{s=1}^t s^{-\alpha}} \\ +4K^2\lambda\tfrac{\tilde\delta\sum_{s=1}^t s^{-\alpha-\beta}}{\sum_{s=1}^t s^{-\alpha}} + \tfrac{\tilde\gamma}{2}\tfrac{\sum_{s=1}^t s^{-2\alpha}\sum_{k=1}^K \|\mathbf{V}_{k,s}\|_*^2}{\sum_{s=1}^t s^{-\alpha}}. \tag{C.15}$$

Since (C.6a) lends $\{Z_t\}$ the quality of a martingale difference sequence, $\mathbb{E}[\sum_{s=1}^t Z_s] = 0$, and (C.8) follows by expectation of (C.15).

(iii) Setting $\alpha = \beta = 0$ in (C.8) gives us (C.9). By using the bounds $\bar v_1, \dots, \bar v_K$ in combination with (C.4) and (C.6b), we find that $|Z_t| \le 4\bar v d\tilde\gamma$ for $t = 1,\dots,T$, and the martingale difference sequence $\{Z_t\}$ is bounded. It follows from Azuma's inequality that $\mathbb{P}(\sum_{t=1}^T Z_t > T\varepsilon\,\tilde\gamma) \le \exp[-\frac{(T\tilde\gamma\varepsilon)^2}{2T(4\bar v d\tilde\gamma)^2}]$ for any $\varepsilon > 0$, which is equivalent to (C.10). ∎

Theorems 1 and 2 follow from Lemmas B.1 and C.1.

Proof of Theorem 1. Following the line of thought of the proof of [43, Theorem 5.1], we first show there one can find a solution $\mathbf{Q}^* \in \mathcal{Q}^*$ such that

$$\liminf_{t\to\infty} \mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) = 0 \quad \text{a.s.}. \tag{C.16}$$

Next, we see that $\{\mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*)\}$ converges almost surely (a.s.) towards a finite quantity which, in view of (C.16), can only be 0. A.s. convergence of $\{\mathbf{Q}_t\}$ towards $\mathbf{Q}^*$ can then be inferred from Lemma A.2-(A.11b). The assumption of non-increasing $\{\delta_t\}$, together with (22a) and (22b), implies $\delta_t \downarrow 0$ which, in view of (B.3), secures a.s. convergence of $\{\hat{\mathbf{Q}}_t\}$ as well.

First observe that (C.16) holds if, almost surely, there exists a subsequence of $\{\mathbf{Q}_t\}$ that converges towards a solution $\mathbf{Q}^* \in \mathcal{Q}^*$. Suppose this condition not to hold, and let $\mathcal{S}$ denote the set of the limit points of all subsequences of $\{\mathbf{Q}_t\}$. Then, almost surely, we have $\mathcal{Q}^* \cap \mathcal{S} = \emptyset$ and, since $\mathcal{S}$ is closed by construction and R is continuous and convex, $\varrho := R^* - \max_{\mathbf{Q}\in\mathcal{S}} R(\mathbf{Q}) > 0$.

Telescoping (C.7) in Lemma C.1(i) and using(B.5), yields

$$\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*) \le \mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*) - \textstyle\sum_{s=1}^t \gamma_s\,[R^* - R(\mathbf{Q}_s)] \\ + \textstyle\sum_{s=1}^t Z_s + 4K^2\lambda\sum_{s=1}^t \gamma_s\delta_s + \frac{K(R^*d)^2}{2^{2K+1}}\sum_{s=1}^t \frac{\gamma_s^2}{\delta_s^2}, \tag{C.17}$$

where $\{Z_t\}$ is the difference sequence of a martingale with respect to the filtration $\{\mathcal{F}_t\}$. In view of (22b) and (22c), the last two terms in the second member of (C.17) converge as $t \to \infty$. As for the third term, since

$$\textstyle\sum_{s=1}^\infty \mathbb{E}[Z_s^2|\mathcal{F}_{s-1}] \le \frac{4K(R^*d)^2}{2^{2K}}\sum_{s=1}^\infty \frac{\gamma_s^2}{\delta_s^2} \overset{(22c)}{<} \infty, \tag{C.18}$$

[44, Theorem 2.18] applies with parameter $p = 2$, and it follows that $\sum_{s=1}^t Z_s$ converges a.s. as $t \to \infty$. Finally, one can find a subsequence $\{\mathbf{Q}_{t_s}\}$ that converges to a point of $\mathcal{S}$ and thus satisfies $R^* - R(\mathbf{Q}_{t_s}) > \varrho/2$ for s large enough. It follows from (22a) that the second term $\sum_{s=1}^\infty \gamma_t[R(\mathbf{Q}_t) - R(\mathbf{Q}^*)] \to -\infty$. All in all we find that $\mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) \to -\infty$ a.s., which is in contradiction with the nonnegativity of $\mathcal{L}$. We infer that (C.16) is true.

It remains to show that $\{\mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*)\}$ is almost surely convergent. To do so we rely on Doob's convergence theorem for supermartingales [44, Theorem 2.5]. Recalling (C.7), and using (B.5) and $R(\mathbf{Q}_t) - R(\mathbf{Q}^*) \le 0$, we find

$$\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*) \le \mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*) + Z_t + 4K^2\lambda\gamma_t\delta_t + \tfrac{K(R^*d\gamma_t)^2}{2^{2K+1}\delta_t^2}. \tag{C.19}$$

Consider $S_t = \sum_{s=t+1}^\infty [4K^2\lambda\gamma_s\delta_s + K(R^*d\gamma_s)^2/(2^{2K+1}\delta_s^2)] + \mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*)$. Under assumptions (22b) and (22c), S_0 is finite by construction. We infer from (C.6a) and (C.19) that $\mathbb{E}[S_t|\mathcal{F}_{t-1}] \le S_{t-1}$ for $t \ge 1$, and $\{S_t\}$ is a supermartingale with respect to $\{\mathcal{F}_t\}$, thus satisfying $\mathbb{E}[S_t] \le S_0 < \infty$. Hence, $\{S_t\}$ is uniformly L^1-bounded and Doob's theorem applies. It follows that $\{S_t\}$, and consequently $\{\mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*)\}$, are almost surely convergent, which completes the proof. ∎

Proof of Theorem 2. By considering Lemma C.1(iii) with the upper bounds $(d\bar v) = dR^*/(2^{K-1}\tilde\delta)$ supplied by (B.5), we find

$$R^* - \mathbb{E}\Big[R(\bar{\mathbf{Q}}_T)\Big] \overset{(\text{C.9})}{\le} \tfrac{K\log M}{T\tilde\gamma} + 4K^2\lambda\tilde\delta + \tfrac{K(R^*d)^2\tilde\gamma}{2^{2K-1}\tilde\delta^2}, \tag{C.20}$$

where we have used $\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*) \le K\log M$, and

$$\mathbb{P}\big(\tfrac{1}{T\tilde\gamma}\textstyle\sum_{t=1}^T Z_t \le \varepsilon\big) \overset{(\text{C.10})}{\ge} 1 - \exp\big(-\tfrac{2^{2K-5}T\varepsilon^2\tilde\delta^2}{(R^*Kd)^2}\big). \tag{C.21}$$

The right member of (C.20) is convex in $(\tilde\gamma,\tilde\delta)$ and minimized for the policy $(\tilde\gamma,\tilde\delta)=(\gamma\,T^{-3/4},\delta\,T^{-1/4})$, where

$$\gamma=\sqrt{\tfrac{2^K}{\lambda R^* K d}}\left(\tfrac{\log M}{2}\right)^{3/4},\quad \delta=2\sqrt{\tfrac{\lambda R^* K^3 d}{2^K}}\left(\tfrac{\log M}{2}\right)^{1/4}. \tag{C.22}$$

We find (28) by substituting $\tilde\gamma$ and $\tilde\delta$ in (C.20) with the suggestion $(\tilde\gamma,\tilde\delta)=(\gamma\,T^{-3/4},\delta\,T^{-1/4})$. Then, (29) follows from (28) and (C.21) after setting $\tilde\delta=\delta\,T^{-1/4}$ in the right member of (C.21). Claims (a) and (b) have been shown. ∎

D. Analysis of the MXL0$^+$ algorithm

The $\mathcal{O}(\delta)$ bound for the bias in Lemma B.1 still holds when the SPSAplus gradient estimator is used. The offset in (SPSA+) allows us, however, to derive an $\mathcal{O}(1/\delta)$ bound for the norm, in place of the harmful $\mathcal{O}(1/\delta)$ bound inherent with SPSA.

Lemma D.1. *If MXL0$^+$ is implemented under* (H0) *and* (H3)-(H4)*, then $\|\mathbf{V}_{k,t}\|_*$ is uniformly bounded for $k=1,\dots,K$.*

In particular, if $(\gamma_t,\delta_t)=(\gamma\,t^{-\alpha},\delta\,t^{-\beta})$ with

$$\text{(a)}\quad 0\le\beta\le\alpha,\qquad\qquad \text{(b)}\quad dLK\,\gamma<2\,\delta, \tag{D.1}$$

then there is $\bar v_{\alpha,\beta}(\gamma,\delta)<\infty$ such that $\|\mathbf{V}_{k,t}\|_*\le\frac{d_k}{2}\,\bar v_{\alpha,\beta}(\gamma,\delta)$ holds for all t and for $k=1,\dots,K$ and, when $\beta=0$,

$$\bar v_{\alpha,0}(\gamma,\delta)=\left(\tfrac{4\tau_\alpha}{\sqrt{d}}\right)\left(\tfrac{2}{dLK}-\tfrac{\gamma}{\delta}\right)^{-1}. \tag{D.2}$$

Proof of Lemma D.1. With the convention $\rho_0=R(\mathbf{Q}_1)$, we have, for $k=1,\dots,K$,

$$\begin{aligned}\|\mathbf{V}_{k,1}\|_*&\overset{\text{(SPSA+)}}{\le}\tfrac{d_k}{\delta_1}|R(\hat{\mathbf{Q}}_1)-R(\mathbf{Q}_1)|\|\mathbf{Z}_{k,1}\|_*\\&\overset{(12)}{\le}\tfrac{d_kL}{2\delta_1}\|\hat{\mathbf{Q}}_1-\mathbf{Q}_1\|\overset{\text{(B.3)}}{\le}d_kLK\sqrt{d},\end{aligned} \tag{D.3}$$

and it follows from (A.2) that $\|\mathbf{V}_1\|_*\le dLK\sqrt{d}$. For $t\ge2$,

$$\begin{aligned}\|\mathbf{V}_{k,t}\|_*&\overset{\text{(SPSA+)}}{\le}\tfrac{d_k}{\delta_1}|R(\hat{\mathbf{Q}}_t)-R(\hat{\mathbf{Q}}_{t-1})|\|\mathbf{Z}_{k,t}\|_*\\&\le\tfrac{d_k}{2\delta_t}\big[|R(\hat{\mathbf{Q}}_t)-R(\mathbf{Q}_t)|+|R(\mathbf{Q}_t)-R(\mathbf{Q}_{t-1})|\\&\qquad+|R(\hat{\mathbf{Q}}_{t-1})-R(\mathbf{Q}_{t-1})|\big]\\&\overset{(12)}{\le}\tfrac{d_k}{2\delta_t}\big[L\|\hat{\mathbf{Q}}_t-\mathbf{Q}_t\|+L\|\mathbf{Q}_t-\mathbf{Q}_{t-1}\|\\&\qquad+L\|\hat{\mathbf{Q}}_{t-1}-\mathbf{Q}_{t-1}\|\big]\\&\overset{\text{(B.3)}}{\le}\tfrac{d_kL}{2\delta_t}\big[2K\sqrt{d}(\delta_t+\delta_{t-1})+\textstyle\sum_{k=1}^K\|\mathbf{Q}_{k,t}-\mathbf{Q}_{k,t-1}\|\big]\\&\overset{\text{(A.9)}}{\le}\tfrac{d_kL}{2\delta_t}\big[2K\sqrt{d}(\delta_t+\delta_{t-1})+K\gamma_t\|\mathbf{V}_{t-1}\|_*\big],\end{aligned} \tag{D.4}$$

so that $\|\mathbf{V}_t\|_*\le\frac{dL}{2\delta_t}[2K\sqrt{d}(\delta_t+\delta_{t-1})+K\gamma_t\|\mathbf{V}_{t-1}\|_*]$. With the convention $\delta_0=0$, we find, by induction on t,

$$\|\mathbf{V}_{k,t}\|_*\le LKd_k\sqrt{d}\textstyle\sum_{s=1}^t(\frac{dLK}{2})^{t-s}\prod_{u=s+1}^t\frac{\gamma_{u-1}}{\delta_u}(1+\frac{\delta_{s-1}}{\delta_s}). \tag{D.5}$$

Condition (H4) tells us that δ_{t-1}/δ_t is uniformly bounded by a finite constant, say, $c<\infty$, while (H3) rewrites as

$$q:=\tfrac{dLK}{2}\,(\sup_{t\ge2}\tfrac{\gamma_{t-1}}{\delta_t})<1. \tag{D.6}$$

Using $\frac{\gamma_{t-1}}{\delta_t}\le\frac{2q}{dLK}$ and $\frac{\delta_{t-1}}{\delta_t}\le c$ in (D.5), we find, for $t\ge2$,

$$\tfrac{\|\mathbf{V}_{k,t}\|_*}{LKd_k\sqrt{d}}\le\textstyle\sum_{s=1}^t q^{t-s}(1+c)=(1+c)\frac{1-q^t}{1-q}\le\frac{1+c}{1-q}. \tag{D.7}$$

Under the policies $\gamma_t=\gamma\,t^{-\alpha}$ and $\delta_t=\delta\,t^{-\beta}$, (D.5) becomes

$$\|\mathbf{V}_{k,t}\|_*\le\tau_\alpha d_kLK\sqrt{d}\textstyle\sum_{s=1}^t\left(\frac{\gamma dLK}{2\delta}\right)^{t-s}\left(\frac{t}{s}\right)^\beta\left[\frac{(t-1)!}{(s-1)!}\right]^{\beta-\alpha}, \tag{D.8}$$

where $\tau_\alpha=1+2^\alpha$. Under Condition (D.1a) the last factor is no larger than 1, and we obtain the uniform bound with

$$\bar v_{\alpha,\beta}(\gamma,\delta)=2L(1+2^\alpha)K\sqrt{d}\textstyle\sum_{s=1}^\infty[\frac{\gamma dLK}{2\delta}]^{t-s}(\frac{t}{s})^\beta, \tag{D.9}$$

which is finite on condition that (D.1b) holds. For $\beta=0$,(D.9) reduces to a geometric series and (D.2) follows directly. ∎

We are now able to show Theorem 3 and Theorem 4. Again, the Lyapunov function (C.1) and Lemma C.1 are used.

Proof of Theorem 3. Proceed as in the proof of Theorem 1, now with assumptions (H1a), (H1b) and (H2) in place of (22a), (22b), (22c). Because the conditions of Lemma D.1 are met, there exists $\bar v<\infty$ such that $\|\mathbf{V}_{k,t}\|_*<\bar v$ for all k, so that (C.17) and (C.19) respectively become, for some $\mathbf{Q}^*\in\mathcal{Q}^*$,

$$\begin{aligned}\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*)&\le\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)-\textstyle\sum_{s=1}^t\gamma_s\,[R^*-R(\mathbf{Q}_s)]\\&\quad+\textstyle\sum_{s=1}^t Z_s+4K^2\lambda\sum_{s=1}^t\gamma_s\delta_s+\frac{K\bar v^2}{2}\sum_{s=1}^t\gamma_s^2,\end{aligned} \tag{D.10}$$

with $\sum_{s=1}^\infty\mathbb{E}\left[Z_s^2|\mathcal{F}_{s-1}\right]\le(4K\bar v)^2\sum_{s=1}^\infty\gamma_s^2<\infty$, and

$$\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*)\le\mathcal{L}(\mathbf{Y}_t;\mathbf{Q}^*)+Z_t+4K^2\lambda\gamma_t\delta_t+\tfrac{K\bar v^2}{2}\gamma_t^2. \tag{D.11}$$

Thus, $S_t=\mathcal{L}(\mathbf{Y}_{t+1};\mathbf{Q}^*)+\sum_{s=t+1}^\infty[4K^2\lambda\gamma_s\delta_s+K\bar v^2\gamma_s^2/2]$ now defines the supermartingale with respect to $\{\mathcal{F}_t\}$. ∎

Proof of Theorem 4. (1) By combining the uniform bound in Lemma D.1 with (C.8) in Lemma C.1(ii) and using $\mathcal{L}(\mathbf{Y}_1;\mathbf{Q}^*)\le K\log M$, we find, for the policy $(\gamma_t,\delta_t)=(\tilde\gamma\,t^{-\alpha},\tilde\delta\,t^{-\beta})$,

$$\begin{aligned}R^*-\mathbb{E}[R(\bar{\mathbf{Q}}_t)]\le\tfrac{K\log M}{\gamma\sum_{s=1}^t s^{-\alpha}}&+4K^2\lambda\delta\tfrac{\sum_{s=1}^t s^{-\alpha-\beta}}{\sum_{s=1}^t s^{-\alpha}}\\&+\tfrac{Kd^2[\bar v_{\alpha,\beta}(\gamma,\delta)]^2\gamma}{8}\tfrac{\sum_{s=1}^t s^{-2\alpha}}{\sum_{s=1}^t s^{-\alpha}},\end{aligned} \tag{D.12}$$

where $\bar v_{\alpha,\beta}(\gamma,\delta)$ is given by (D.9). The above upper bound is minimized for $\alpha=\beta=1/2$, in which case we find (27).

(2) Using Lemma C.1(iii) under $(\tilde\gamma,\tilde\delta)=(\gamma/\sqrt{T},\delta/\sqrt{T})$ and with the bounds $\bar v=\frac12\,\bar v_{0,0}(\tilde\gamma,\tilde\delta)$, given by Lemma D.1, yields

$$R^*-\mathbb{E}[R(\bar{\mathbf{Q}}_T)]\overset{\text{(C.9)}}{\le}\tfrac{K\log M}{\gamma\sqrt{T}}+\tfrac{4K^2\lambda\delta}{\sqrt{T}}+\tfrac{Kd^2[\,\bar v_{0,0}(\frac{\gamma}{\sqrt{T}},\frac{\delta}{\sqrt{T}})]^2\gamma}{8\sqrt{T}} \tag{D.13}$$

where $\bar v_{0,0}(\tilde\gamma,\tilde\delta)=(\frac{8}{\sqrt{d}})(\frac{2}{dLK}-\frac{\gamma}{\delta})^{-1}$, and

$$\mathbb{P}\Big(\tfrac{\sum_{t=1}^T Z_t}{\sqrt{T}\,\gamma}\le\varepsilon\Big)\overset{\text{(C.10)}}{\ge}1-\exp\Big(-\tfrac{T\varepsilon^2}{8K^2d^2[\,\bar v_{0,0}(\frac{\gamma}{\sqrt{T}},\frac{\delta}{\sqrt{T}})]^2}\Big). \tag{D.14}$$

We find (23) after substituting $\bar v_{0,0}(\tilde\gamma,\tilde\delta)$ with its actual value in (D.13). Then, (24) follows from (23) and (D.14). ∎

The proof of Corollary 1 relies on the following lemma.

Lemma D.2. *Let $\Gamma=\{(\gamma,\delta)\in\mathbb{R}_{>0}\times\mathbb{R}_{>0}:\gamma<h\delta\}$ and consider the function $f:\Gamma\mapsto\mathbb{R}$ defined by*

$$f(\gamma,\delta)=\tfrac{a}{\gamma}+2b\delta+c\gamma\,(h-\tfrac{\gamma}{\delta})^{-2}+d\,(h-\tfrac{\gamma}{\delta})^{-1}, \tag{D.15}$$

where $a,b,c,h>0$ and $d\ge0$ are given parameters.

(i) At the point $(\gamma^*,\delta^*)\in\Gamma$, where

$$\gamma^*=h\left[\sqrt{\tfrac{c}{a}}+\sqrt{\sqrt{\tfrac{c}{a}}\left(\tfrac{2bh}{2\sqrt{ac}+d}\right)}\right]^{-1},\quad \delta^*=\sqrt{\sqrt{\tfrac{a}{c}}\left(\tfrac{2\sqrt{ac}+d}{2bh}\right)}, \tag{D.16}$$

the value of f is given by

$$f(\gamma^*,\delta^*)=\tfrac{2\sqrt{ac}+d}{h}+2\sqrt{2b\sqrt{\tfrac{a}{c}}\left(\tfrac{2\sqrt{ac}+d}{h}\right)}. \tag{D.17}$$

Under the constraint $\delta/\sqrt{t} < r$*, where* $r > 0$*,* (D.17) *holds for*

$$t > \sqrt{\tfrac{a}{c}}\left(\tfrac{2\sqrt{ac}+d}{2bh}\right) r^{-2}. \tag{D.18}$$

(ii) For any $\varepsilon > 0$*,* $f(\gamma^*, \delta^*)/\sqrt{t} \le \varepsilon$ *holds for* $t \ge T$ *if* $T = [f(\gamma^*, \delta^*)/\varepsilon]^2$*. The constraint* $\delta^*/\sqrt{T} < r$ *then rewrites as*

$$\varepsilon < \left(4b + \sqrt{2(b/h)\sqrt{c/a}\,(2\sqrt{ac}+d)}\right) r. \tag{D.19}$$

Proof. Verification of all the claims is straightforward. ∎

Proof of Corollary 1. (a) To derive γ and δ in (a) it suffices to apply Lemma D.2(i) to the expression for $B(\gamma, \delta)$ given in Theorem 4(2a). The convergence rate of $\mathbb{E}[R(\bar{\mathbf{Q}}_T)]$ follows from (28) and (D.17), while the condition on T is a translation of (D.18) into the present setting, where the restriction $\delta/\sqrt{s} < r_k$ for all k applies, with r_k given by (21).

(b) Recall Theorem 4(2b). The second part of (29) rewrites as $1-\alpha$ for $\varepsilon = 16\left[\frac{2}{LK(M^2-1)} - \frac{\gamma}{\delta}\right]^{-1}\sqrt{2\log(\frac{1}{\alpha})K^2(M^2-1)/T}$. Observe that $B(\gamma,\delta)+\varepsilon\sqrt{T}$ is an instance of the function $f(\gamma,\delta)$ defined in (D.15). Lemma D.2(ii) gives us a condition on T for $B(\gamma,\delta)/\sqrt{\lceil T\rceil} + \varepsilon \le \varepsilon$ to be true which, in view of (29), is also sufficient for (31) to hold. After computations we find the value of T in Table IIb with the restriction on ε:

$$\varepsilon \overset{(D.19)}{<} 2^{\frac{9}{4}}\lambda\left[\phi(\alpha)\sqrt{L/\lambda} + \frac{2^{\frac{3}{4}}}{\sqrt[4]{\log(1/\alpha)[M^2-1]}}\right] \times \sqrt{\frac{\sqrt{\log(1/\alpha)}K^4[M+1][M^2-1]}{M}}. \quad ∎$$

E. Analysis of the AMXL0$^+$ algorithm

Lemmas B.1 and D.1 still apply in the asynchronous setting. Instead of (C.1) we use the Lyapunov function

$$\mathcal{L}_\pi(\mathbf{Y};\mathbf{Q}^*) = \textstyle\sum_{k=1}^K \frac{1}{\pi_k} F(\mathbf{Q}_k^*, \mathbf{Y}_k), \tag{E.1}$$

where $\mathbf{Q}^* \in \mathcal{Q}^*$ is a solution. Proceeding as for the derivation of (C.12) in Lemma C.1, we find, for the algorithm (AMXL0$^+$),

$$\begin{aligned}\mathcal{L}_\pi(\mathbf{Y}_{t+1};\mathbf{Q}^*) \le \mathcal{L}_\pi(\mathbf{Y}_t;\mathbf{Q}^*) - \gamma_t\,[R^* - R(\mathbf{Q}_t)] \\ +4K^2\lambda\gamma_t\delta_t + X_t + \textstyle\sum_{k=1}^K \frac{\gamma_t^2}{2}\|\mathbf{V}_{k,t}\|_*^2,\end{aligned} \tag{E.2}$$

with the random sequence $\{X_t\}$ now given by

$$\begin{aligned}X_t = \gamma_t \textstyle\sum_{k\in U_t}\langle \mathbf{U}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle \\ + \textstyle\sum_{k=1}^K \frac{\mathbf{1}_{U_t}(k)-\pi_k}{\pi_k}\,[\gamma_t\langle \mathbf{V}_{k,t}, \mathbf{Q}_{k,t} - \mathbf{Q}_k^*\rangle + \frac{\gamma_t^2}{2}\|\mathbf{V}_{k,t}\|_*^2].\end{aligned} \tag{E.3}$$

It is easily seen that $\mathbb{E}[X_t|\mathcal{F}_{n-1}] = 0$, and

$$\begin{aligned}|X_t| \le 2\gamma_t \textstyle\sum_{k=1}^K \|\mathbf{U}_{k,t}\|_* \\ + \textstyle\sum_{k=1}^K \max\left(1, \frac{1}{\pi_k} - 1\right)[2\gamma_t\|\mathbf{V}_{k,t}\|_* + \frac{\gamma_t^2}{2}\|\mathbf{V}_{k,t}\|_*^2].\end{aligned} \tag{E.4}$$

Compare (E.2),(E.4) with (C.7),(C.6b). By reproducing the rationale behind the proof of Lemma C.1, we obtain an asynchronous counterpart to Lemma C.1, where (C.8) and (C.9) now hold with $\mathcal{L}_\pi$ in place of $\mathcal{L}$, and (C.10) becomes

$$\mathbb{P}\Big(\tfrac{\sum_{t=1}^T X_t}{T\tilde{\gamma}} \ge \zeta\Big) \le \exp\Big(-\frac{T\zeta^2}{8K^2d^2\left[(1+\frac{\upsilon_\pi}{2})\bar{v}_{0,0}(\tilde{\gamma},\tilde{\delta}) + \frac{\upsilon_\pi}{8}\tilde{\gamma}d\,[\bar{v}_{0,0}(\tilde{\gamma},\tilde{\delta})]^2\right]^2}\Big), \tag{E.5}$$

where υ_π is defined as in Theorem 5.

Proof of Theorem 5. Proceed as in the proof of Theorem 4. ∎

Proof of Corollary 2. First observe that we have $\upsilon_\pi = K-1$ if $K \ge 2$. The rest of the proof bases on the conclusions of Theorem 5 and follows the exact lines of the proof of Corollary 1, now using $B_\pi(\gamma,\delta)$ and $C_\pi(\gamma,\delta)$. Note that Corollary 2b holds with the following restriction on ε:

$$\varepsilon \overset{(D.19)}{<} 2^3\lambda\Big[\hat{\phi}_\pi(\alpha)\sqrt{\hat{\chi}(\alpha)L/\lambda} + \left[\log^{\frac{3}{8}}(1/\alpha)K^{\frac{3}{4}}d\right]^{-1}\Big] \times \sqrt{\frac{\log^{\frac{3}{4}}(1/\alpha)K^{\frac{11}{4}}[M+1][M^2-1]}{M}}. \quad ∎$$

References

[1] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, February 2014.

[2] J. G. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[3] J. S.-B. Orange, A. G. Armada, B. Evans, A. Galis, and H. Karl, "White paper for research beyond 5G," *Accessed*, vol. 23, pp. 1–43, 2015.

[4] E. Calvanese Strinati, S. Barbarossa, J. L. Gonzalez-Jimenez, D. Ktenas, N. Cassiau, L. Maret, and C. Dehos, "6g: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 42–50, 2019.

[5] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?" *IEEE Trans. Wireless Commun.*, vol. 31, no. 2, pp. 160–171, February 2013.

[6] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, pp. 40–60, January 2013.

[7] R. S. Cheng and S. Verdú, "Gaussian multiaccess channels with ISI: capacity region and multiuser water-filling," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 773–785, May 1993.

[8] W. Yu, W. Rhee, S. Boyd, and J. M. Cioffi, "Iterative water-filling for Gaussian vector multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 1, pp. 145–152, 2004.

[9] G. Scutari, D. P. Palomar, and S. Barbarossa, "Optimal linear precoding strategies for wideband non-cooperative systems based on game theory – part I: Nash equilibria," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1230–1249, March 2008.

[10] ——, "Optimal linear precoding strategies for wideband non-cooperative systems based on game theory – part II: algorithms," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1250–1267, March 2008.

[11] E. V. Belmega, S. Lasaulce, M. Debbah, M. Jungers, and J. Dumont, "Power allocation games in wireless networks of multi-antenna terminals," *Telecommunication Systems*, vol. 47, no. 1-2, pp. 109–122, 2011.

[12] G. Scutari, D. P. Palomar, and S. Barbarossa, "Asynchronous iterative waterfilling for Gaussian frequency-selective interference channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 2868–2878, July 2008.

[13] ——, "Simultaneous iterative water-filling for Gaussian frequency-selective interference channels," in *ISIT '06: Proceedings of the 2006 International Symposium on Information Theory*, 2006.

[14] E. Hosseini and A. Falahati, "Improving water-filling algorithm to power control cognitive radio system based upon traffic parameters and QoS," *Wireless Personal Communications*, vol. 70, pp. 1747–1759, 2013.

[15] S. S. Christensen, R. Agarwal, E. De Carvalho, and J. M. Cioffi, "Weighted sum-rate maximization using weighted MMSE for MIMO-BC beamforming design," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4792–4799, 2008.

[16] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4331–4340, 2011.

[17] P. Mertikopoulos and A. L. Moustakas, "Learning in an uncertain world: MIMO covariance matrix optimization with imperfect feedback," *IEEE Trans. Signal Process.*, vol. 64, no. 1, pp. 5–18, January 2016.

[18] G. Scutari, D. P. Palomar, and S. Barbarossa, "The MIMO iterative waterfilling algorithm," *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 1917–1935, May 2009.

[19] R. Liao, B. Bellalta, M. Oliver, and Z. Niu, "MU-MIMO MAC protocols for wireless local area networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 162–183, January 2016.

[20] J. C. Spall, "A one-measurement form of simultaneous perturbation stochastic approximation," *Automatica*, vol. 33, no. 1, pp. 109–112, 1997.

[21] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, "Online convex optimization in the bandit setting: gradient descent without a gradient," in *SODA '05: Proceedings of the 16th annual ACM-SIAM Symposium on Discrete Algorithms*, 2005, pp. 385–394.

[22] W. Li and M. Assaad, "Matrix exponential learning schemes with low informational exchange," *IEEE Trans. Signal Process.*, vol. 67, no. 12, pp. 3140–3153, April 2019.

[23] T. Dahl, N. Christophersen, and D. Gesbert, "Blind MIMO eigenmode transmission based on the algebraic power method," *IEEE Trans. Signal Process.*, vol. 52, no. 9, pp. 2424–2431, September 2004.

[24] D. Ogbe, D. J. Love, and V. Raghavan, "Noisy beam alignment techniques for reciprocal MIMO channels," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 5092–5107, October 2017.

[25] N. Jindal, S. Vishwanath, and A. Goldsmith, "On the duality of Gaussian multiple-access and broadcast channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 5, pp. 768–783, 2004.

[26] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications and Related Technologies*, vol. 10, no. 6, pp. 585–596, 1999.

[27] K. Senel, H. V. Cheng, E. Björnson, and E. G. Larsson, "What role can NOMA play in massive MIMO?" *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 3, pp. 597–611, 2019.

[28] D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124 – 143, 1996.

[29] A. Neyman, "Correlated equilibrium and potential games," *International Journal of Game Theory*, vol. 26, no. 2, pp. 223–227, June 1997.

[30] Z.-Q. Luo and J.-S. Pang, "Analysis of iterative waterfilling algorithm for multiuser power control in digital subscriber lines," *EURASIP J. Adv. Signal Process*, pp. 1–10, Jan. 2006.

[31] P. Mertikopoulos, E. V. Belmega, A. L. Moustakas, and S. Lasaulce, "Distributed learning policies for power allocation in multiple access channels," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 1, pp. 96–106, January 2012.

[32] P. Mertikopoulos, E. V. Belmega, and A. L. Moustakas, "Matrix exponential learning: Distributed optimization in MIMO systems," in *ISIT '12: Proceedings of the 2012 IEEE International Symposium on Information Theory*, 2012, pp. 3028–3032.

[33] Y. Nesterov, "Primal-dual subgradient methods for convex problems," *Mathematical Programming*, vol. 120, no. 1, pp. 221–259, 2009.

[34] L. Liu, C. Oestges, J. Poutanen, K. Haneda, P. Vainikainen, F. Quitin, F. Tufvesson, and P. D. Doncker, "The COST 2100 MIMO channel model," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 92–99, December 2012.

[35] COST Action 231, "Digital mobile radio towards future generation systems," European Commission, final report, April 1999.

[36] L. Sanguinetti, E. Björnson, and J. Hoydis, "Toward massive MIMO 2.0: Understanding spatial correlation, interference suppression, and pilot contamination," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 232–257, 2019.

[37] P. Mertikopoulos, E. V. Belmega, R. Negrel, and L. Sanguinetti, "Distributed stochastic optimization via matrix exponential learning," *IEEE Trans. Signal Process.*, vol. 65, no. 9, pp. 2277–2290, May 2017.

[38] P. Mertikopoulos and E. V. Belmega, "Learning to be green: Robust energy efficiency maximization in dynamic MIMO-OFDM systems," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 743 – 757, April 2016.

[39] Y.-L. Yu, "The strong convexity of von Neumann's entropy," June 2013, unpublished note. [Online]. Available: http://www.cs.cmu.edu/~yaoliang/mynotes/sc.pdf

[40] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2014. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611973433

[41] S. M. Kakade, S. Shalev-Shwartz, and A. Tewari, "Regularization techniques for learning with matrices," *J. Mach. Learn. Res.*, vol. 13, pp. 1865–1890, Jun. 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2188385.2343703

[42] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 1st ed. Springer Publishing Company, Incorporated, 2014.

[43] M. Bravo, D. S. Leslie, and P. Mertikopoulos, "Bandit learning in concave N-person games," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS'18, 2018, pp. 5666–5676.

[44] P. Hall and C. C. Heyde, *Martingale limit theory and its application / P. Hall, C.C. Heyde*. Academic Press New York, 1980.